ELSEVIER

Theoretical Computer Science 172 (1997) 135–174

Theoretical Computer Science

Deterministic summation modulo $\mathscr{B}_n$, the semigroup of binary relations on $\{0, 1, \ldots, n-1\}$

W.G. Handley[1]

BUCS, University of Bath, Bath, BA2 7AY, UK

Received May 1994
Communicated by R.V. Book

Abstract

Deterministic summation *modulo* an arbitrary semigroup generalises the schemes of bounded summation and bounded product used to build up the class of elementary functions. As a special case it includes bounded first-order quantification over the integers. S_n, as usual, is the group of permutations of $[n] = \{0, 1, \ldots, n-1\}$. $\mathscr{B}_n$ stands for the monoid of all binary relations on $[n]$, with relational composition as the semigroup operation. We show that in the presence of bounded quantification and boolean operations, deterministic summation *modulo* the group S_n is already as powerful as deterministic summation *modulo* the monoid $\mathscr{B}_n$. The computational significance is that for certain machines (Bel'tyukov's stack register machines), the first corresponds to determinism, the second to nondeterminism.

1. Introduction

$\Delta_0^{\mathbf{N}}$ is the class of sets of the form

$$\{\vec{x} : \underbrace{(\exists z_1 \leqslant p_1(\vec{x}))(\forall z_2 \leqslant p_2(\vec{x})) \cdots (\forall_{\exists} z_k \leqslant p_k(\vec{x}))}_{\text{finitely many alternating bounded quantifiers}} [q(\vec{z}, \vec{x}) = 0]\}$$

where $p_i(\vec{x}) \in \mathbb{N}[\vec{x}]$ and $q(\vec{z}, \vec{x}) \in \mathbb{Z}[\vec{z}, \vec{x}]$. Smullyan called these sets the constructive arithmetic predicates [19] and gave Definition 8 (below, in Section 2), which is equivalent and more easily extended.

Although there are classes of formulae (studied for example in [5]) which relate more directly to Turing machine computations, Δ_0-formulae are beautifully tailored to expressing familiar properties of numbers, including ones which are relevant to the $\mathrm{P} \stackrel{?}{=} \mathrm{NP}$ question. The set of prime numbers and, since $\Delta_0^{\mathbf{N}}$ is closed under comple-

[1] Engineering and Physical Sciences Research Council – Research Fellowship B/92/RF/1522. Revised while employed by University of Wales Swansea.

0304-3975/97/$17.00 © 1997 — Elsevier Science B.V. All rights reserved
SSDI 0304-3975(95)00245-6

mentation, the set of composite numbers are both in $\Delta_0^{\mathbf{N}}$. So the problems addressed here bear on the big questions of complexity theory.

However, it is remarkable that whereas some computationally demanding sets are known to be in $\Delta_0^{\mathbf{N}}$ the same is not known for closely related sets. To give a couple of examples, let

$$p_0, p_1, p_2, \ldots$$

be the set of primes written out in ascending order. This set is in $\Delta_0^{\mathbf{N}}$. However it is not know whether the relation

$$\{\langle x, y\rangle : x = p_y\}$$

is in $\Delta_0^{\mathbf{N}}$. Worse still, we do not know whether

$$\{p_0, p_2, p_4, \ldots\} \in \Delta_0^{\mathbf{N}}. \tag{1}$$

Nevertheless, the existence of alternative characterizations suggests very strongly that $\Delta_0^{\mathbf{N}}$ is a class of natural significance.

The first alternative characterization is as the class RUD of rudimentary sets, again presented in [19]. Bennett proved in [4] (or see [14]) that RUD $= \Delta_0^{\mathbf{N}}$.

The second alternative is the linear time hierarchy (LTH) defined by Wrathall:

$$\Sigma_0^L = \text{DLINTIME}$$

$$\Sigma_{i+1}^L = \left\{ B \subseteq \{0,1\}^* : (\exists A \in \Sigma_i^L) \left[\begin{array}{l} B \text{ is computable by} \\ \text{a nondeterministic TM} \\ \text{running in linear time} \\ \text{with oracle } A \end{array} \right] \right\}$$

$$LTH = \bigcup_{i \in \mathbf{N}} \Sigma_i^L.$$

Wrathall showed that LTH = RUD in [20]. We write $\Delta_0^{\mathbf{N}}$ in preference over LTH or RUD, but in all the contexts we meet the three are interchangeable.

A third alternative comes with the stack register machines (SRMs) of Bel'tyukov. We postpone a definition of these machines (and the complexity classes based on them) until Section 7 (Definition 13), for although they also provide the principal application of our main theorem (Theorem 1 in Section 3), the latter is entirely independent of them. SRMs are defined in [3], where it is observed that RUD is exactly

$$\text{SRM}[+, \cdot]\left(n^{o(1)}, 0\right),$$

which denotes the class of sets recognised by SRMs

- with oracles for the relations $x + y = z$ and $x \cdot y = z$, and
- running in polynomial space (equivalently polynomial time), but
- not using the special "working" register.

Summation

The definition of elementary functions, due to Kalmar and Csillag (see e.g. [18]), relies, in addition to certain initial functions such as successor, on the schemes of bounded summation and bounded product:

$$\sum_{t \leqslant y} f(t,\vec{x}) = f(0,\vec{x}) + f(1,\vec{x}) + \cdots + f(y,\vec{x}), \tag{2}$$

$$\prod_{t \leqslant y} f(t,\vec{x}) = f(0,\vec{x}) \cdot f(1,\vec{x}) \cdots f(y,\vec{x}). \tag{3}$$

Both these schemes have the same form but the first has addition on the natural numbers as its underlying operation whereas the other has multiplication. One may ask, What happens if one takes not $\mathbb{N}$ but some other set (whose elements are representable by numbers) and not addition or multiplication but some other operation, preferably associative? Take, say, the group $\mathbb{Z}_2$, which is the set $\{0,1\}$ under the operation of addition *modulo* 2: one has an obvious notion of summation (or *counting*) *mod* $\mathbb{Z}_2$; it just gives the parity of the bounded sums

$$f(0,\vec{x}) \oplus f(1,\vec{x}) \oplus \cdots \oplus f(y,\vec{x}).$$

In raising question (1) above, one is asking whether $\Delta_0^{\mathbf{N}}$ is rich enough to handle operations such as summation *mod* $\mathbb{Z}_2$. (In our notation, whether $\mathbb{Z}_2$-$\Delta_0^{\mathbf{N}} = \Delta_0^{\mathbf{N}}$.) Such questions are relevant to models of $I\Delta_0$ (Peano arithmetic with the induction scheme holding only for bounding arithmetic formulae; see [15, 16]).

The results in this paper, it must be said, do not bear directly on the extent of $\Delta_0^{\mathbf{N}}$ itself. Rather they show the power of bounded quantification when combined with other forms of summation. If G is a semigroup,[2] $\mathbb{Z}_2$ for example, add closure under summation *mod* G, call the resulting class G-$\Delta_0^{\mathbf{N}}$, and investigate what else may have been sumggled in.

Semigroups of binary relations

Let $[n]$ be a set with n elements. In places, this paper's title for one, we assume that

$$[n] = \{0,1,2,\ldots,n-1\}$$

but any n-element set will do.

The semigroups of interest for this paper (all in fact are monoids) contain relations and functions over $[n]$.

Definition 1. For a relation $R \subseteq [n] \times [n']$, the domain and range of R are defined by

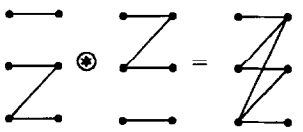
$$\mathrm{Dom}(R) = \{x \in [n] : \exists y \in [n'].\ \langle x,y\rangle \in R\},$$
$$\mathrm{Ran}(R) = \{y \in [n'] : \exists x \in [n].\ \langle x,y\rangle \in R\}.$$

[2] A *semigroup* is a set with a binary associative operation. A *monoid* is a semigroup with a two-sided identity.

Definition 2. If $R \subseteq [n] \times [n']$ and $R' \subseteq [n'] \times [n'']$ then the composite $R \circledast R'$ is defined by

$$R \circledast R' = \{\langle x,z\rangle : \exists y.\ \langle x,y\rangle \in R \wedge \langle y,z\rangle \in R'\}.$$

For example, with $n = n' = n'' = 3$:



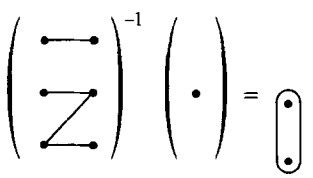
We note that when R is a set of functions, $f \circledast g$ is more usually written $g \circ f$.

The next definition will be useful later.

Definition 3. For $R \subseteq [n] \times [n']$ and $y \in [n']$, the inverse image of y is defined by

$$R^{-1}(y) = \{x : \langle x,y\rangle \in R\}.$$

For example, with $n = n' = 3$:



Definitions 4. (a) $\mathscr{B}_n$ stands for the set of binary relations[3] over $[n]$, i.e. $\mathscr{B}_n = \wp([n] \times [n])$, the set of subsets of $[n] \times [n]$.

(b) $\mathscr{T}_n$ stands for the set of functions with domain and codomain (but not necessarily range) $[n]$. (Other names for $\mathscr{T}_n$ are $([n] \to [n])$ and M_n, the latter in [8].)

(c) $\mathscr{T}_n^\sharp$ stands for the set of *pre-graphs* or *multivalued functions* over $[n]$: that is binary relations with domain the whole of $[n]$, so that

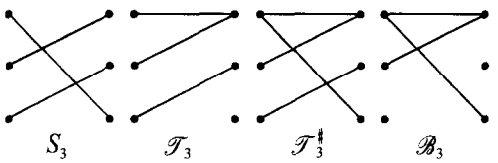
$$R \in \mathscr{T}_n^\sharp \text{ if and only if } (\forall x \in [n])(\exists y \in [n])(\langle x,y\rangle \in R).$$

(Note that y is not necessarily unique; if it is then $R \in \mathscr{T}_n$.)

(d) S_n is the usual group of permutations of $[n]$.

[3] I follow [7] in writing $\mathscr{B}_n$ and $\mathscr{T}_n$

Typical elements are



Clearly $S_n \subsetneq \mathscr{T}_n \subsetneq T_n^\sharp \subsetneq \mathscr{B}_n$ ($\neq$ holding for all $n \geqslant 2$), because a function is a special kind of binary relation and functional composition is a special case of relational composition.

Closure under summation

The chief feature of $\Delta_0^{\mathbf{N}}$ is its closure under bounded quantification and boolean operations (see Definition 6 in Section 2). The class formed by adding closure under, for example, S_n, we denote S_n-$\Delta_0^{\mathbf{N}}$ (see Definition 11).

We take summation *mod* S_n as our base point because these groups arise directly when considering SRMs (see Fact 1 in Section 7). It is trivial that

$$S_2\text{-}\Delta_0^{\mathbf{N}} = \mathbb{Z}_2\text{-}\Delta_0^{\mathbf{N}}$$

and [15] it is known too that, for example,

$$S_3\text{-}\Delta_0^{\mathbf{N}} = \mathbb{Z}_6\text{-}\Delta_0^{\mathbf{N}} = \mathbb{Z}_2\mathbb{Z}_3\text{-}\Delta_0^{\mathbf{N}}.$$

So these closures are, or are very nearly, the smallest possible.

In this paper, we show (Corollary 1 in Section 3) that for all $n \in \mathbb{N}$

$$\mathscr{B}_n\text{-}\Delta_0[\mathscr{F}] = S_n\text{-}\Delta_0[\mathscr{F}], \tag{4}$$

where $\mathscr{F}$ indicates possible relativisation to an oracle.

In [8], by methods which this paper extends, Clote gave direct proofs of (4) for $n \leqslant 2$. For $n \geqslant 5$, he built on the work of Barrington [2] and gave a less direct proof (Fact 2 of Section 7). Indeed, as an acquaintance with Barrington's work might lead one to hope, the result is much more striking: an apparent hierarchy collapses and, for all $n \geqslant 5$,

$$S_n\text{-}\Delta_0^{\mathbf{N}} = \text{ALINTIME}$$

where ALINTIME is the class of relations which can be recognised in linear time by alternating Turing machines (studies by Chandra et al. in [6]).

In this paper we give a direct proof of (4), for all $n \in \mathbb{N}$, including the missing cases $3 \leqslant n \leqslant 4$ (Corollary 1), but we cannot claim fresh insight into other questions of separation/collapse.

Stack register machines

Definition 15 details complexity classes SRM $[\mathscr{F}](\Phi,\Gamma)$ for SRMs. Here $\mathscr{F}$ is a collection of oracles, Γ gives bounds on the special, "working" register, and Φ gives bounds on all the other register. In Definition 19, we pass on to nondeterministic SRMs (NSRMs) and classes NSRM $[\mathscr{F}](\Phi,\Gamma)$.

We know from [15] that

$$\mathrm{SRM}\,[+,\cdot]\left(n^{o(1)},k\right) = S_{k+1}\text{-}\Delta_0^{\mathbf{N}}$$

and we can now show (Corollary 3) that, for all $k \in \mathbb{N}$,

$$\mathrm{NSRM}\,[+,\cdot]\left(n^{o(1)},k\right) = \mathrm{SRM}\,[+,\cdot]\left(n^{o(1)},k\right) = S_{k+1}\text{-}\Delta_0^{\mathbf{N}},$$

which was already proved by Clote in the cases $k \leqslant 1$ and $k \geqslant 4$. Indeed (Corollary 4)

$$\mathrm{NSRM}\,[\mathscr{F}]\left(n^{o(1)},k\right) = \mathrm{SRM}\,[\mathscr{F}]\left(n^{o(1)},k\right) = S_{k+1}\text{-}\Delta_0[\mathscr{F}]$$

for many classes of functions $\mathscr{F}$.

2. Preliminaries

Let $\mathbb{N}$ be the set $\{0,1,2,\ldots\}$ of natural numbers. Let G be a semigroup with semigroup operation $\circledast$.

Definition 4. If G is a semigroup and $F : \mathbb{N}^{(1+k)} \to G$ then

$$\overline{F} : \mathbb{N}^{(1+k)} \to G$$

is defined by

$$\overline{F}(y,\vec{x}) = F(0,\vec{x}) \circledast F(1,\vec{x}) \circledast \cdots \circledast F(y,\vec{x})$$

where $\vec{x} = x_1,\ldots,x_k$ are parameters. Since parameters make no difference to our results and will be clear from the context, we henceforth omit them, writing simply $F : \mathbb{N} \to G$.

For an example of $\overline{F}$, where $F : \mathbb{N} \to T_3^{\sharp}$, the reader may consult Figs. 1 and 2 in Section 3.

Example. Let $\mathbb{N}_{\times}$ be the semigroup whose elements are natural numbers and whose operation is the usual addition. Deterministic closure under summation *mod* $\mathbb{N}_{+}$ gives exactly the bounded summation at (2) above.

Let $\mathbb{N}_{\times}$ be the semigroup whose elements are natural numbers and whose operation is the usual multiplication. Deterministic closure under summation *mod* $\mathbb{N}_{\times}$ gives exactly the bounded product at (3) above.

For the remainder of this paper, we shall assume that G is finite.

Δ_0-closure

In the manner of [19, Ch. II, Part A, Section 1], we say that:

Definition 6. A relation $R \subseteq \mathbb{N}^m$ is Δ_0-definable from existing relations R_1, R_2 if one of the following holds:

(a) *Boolean operations*: $R_1, R_2 \subseteq \mathbb{N}^m$ and $R = R_1 \cup R_2$, $R = R_1 \cap R_2$ or $R = \mathbb{N}^m \backslash R_1$.

(b) *Bounded quantification*: $R_1 \subseteq \mathbb{N}^{m+1}$ and

$$R = \{\langle x_1, \ldots, x_m\rangle : (\exists z \leqslant x_i)\langle z, x_1, \ldots, x_m\rangle \in R_1\}$$

or

$$R = \{\langle x_1, \ldots, x_m\rangle : \forall z \leqslant x_i)\langle z, x_1, \ldots, x_m\rangle \in R_1\}$$

for given $1 \leqslant i \leqslant m$.

(c) *Explicit transformation*: $R_1 \subseteq \mathbb{N}^\ell$ and

$$R = \{\langle x_1, \ldots, x_m\rangle : \langle \xi_1, \ldots, \xi_\ell\rangle \in R_1\}$$

where each ξ_i is an x_j or a natural number.

Definition 7. A class of relations $\mathscr{C}$ over the natural numbers $\mathbb{N}$ is Δ_0-*closed* if

- it contains the binary relations $\{\langle i,j\rangle : i = j\}$ and $\{\langle i,j\rangle : i \leqslant j\}$ and
- if R is definable from $R_1, R_2 \in \mathscr{C}$ by a boolean operation, by bounded quantification, or by explicit transformation then $R \in \mathscr{C}$.

Definition 8. We write $\Delta_0[\mathscr{F}]$ to indicate the closure of $\mathscr{F}$ under Δ_0-definability. That is to say the smallest class $\mathscr{D}$ of relations on $\mathbb{N}$ such that

[•] $\mathscr{F} \subseteq \mathscr{D}$,

[•] $\mathscr{D}$ is Δ_0-closed.

We write $\Delta_0^{\mathbf{N}}$ for $\Delta_0(\{+,\cdot\})$, where $+$ and $\cdot$ are the 3-place relations $\{\langle i,j,k\rangle : i+j = k\}$ and $\{\langle i,j,k\rangle : i \cdot j = k\}$ respectively.

Remark. Δ_0-definability is the *constructive definability* of Smullyan, and $\Delta_0^{\mathbf{N}}$ is the class of *constructive arithmetic* predicates. RUD is $\Delta_0(\{C\})$, where $C(x,y,z)$ iff the dyadic representation of z is identical to the dyadic representations of x and y concatenated. (Each natural number has a unique dyadic representation as a string of 1's and 2's. For example, 5 is represented as 21 because $5 = 2 \cdot 2^1 + 1 \cdot 2^0$. This avoids the problems of leading 0's which a binary representation would cause.) All this is in [19], with further details on RUD in [21].

Closure under summation

Definition 9. For finite G, if $F : \mathbb{N}^m \to G$ then we write

$$F \in \mathscr{C}$$

to mean that, for all $\alpha \in G$,

$$F^{-1}(\alpha) \in \mathscr{C}.$$

Definition 10. *$\mathscr{C}$ is closed under summation mod G* if, for all m, for all $F : \mathbb{N}^m \to G$

$$F \in \mathscr{C} \text{ implies } \overline{F} \in \mathscr{C}.$$

Remark. If $B_\vee$ is the semigroup $\{\underline{True}, \underline{False}\}$, with disjunction as the semigroup operation, then summation *mod* $B_\vee$ is the same as bounded existential quantification.

If $B_\wedge$ is the semigroup $\{\underline{True}, \underline{False}\}$, with conjunction as the semigroup operation, then summation *mod* $B_\wedge$ is the same as bounded universal quantification.

Definition 11. We write $G\text{-}\Delta_0[\mathscr{F}]$ to indicate the closure of $\mathscr{F}$ under Δ_0-operations and summation *mod* G; that is, it is the smallest class $\mathscr{D}$ of relations on $\mathbb{N}$ such that

- $\mathscr{F} \subseteq \mathscr{D}$,
- $\mathscr{D}$ is Δ_0-closed,
- $\mathscr{D}$ is under summation *mod* G.

Formulae and definitions of functions

For ease of exposition, we represent relations by formulae. We now give constructions on formulae to correspond to some of the closure properties above.

If $\mathscr{C}$ is Δ_0-closed then we can use the following constructions:

(a) If $R \subseteq \mathbb{N}^\ell$ is in $\mathscr{C}$ then any formula

$$\langle \xi_1, \ldots, \xi_\ell \rangle \in R$$

also represents a member of $\mathscr{C}$, with $\xi_1, \ldots, \xi_\ell$ as for explicit transformations. R itself is captured by taking $\xi_i = x_i$.

(b) By assumption $x = y$ and $x \leqslant y$ also represent relations in $\mathscr{C}$.

(c) If Φ_1, Φ_2 represent relations known to be in $\mathscr{C}$ then the formulae

$$\Phi_1 \wedge \Phi_2, \quad \Phi_1 \vee \Phi_2 \text{ and } \neg \Phi_1$$

correspond to the boolean operations of $\cap, \cup$ and $\mathbb{N}^m \setminus$ respectively. Extending the first two, we also allow

$$\bigwedge_{i \in I} \Phi_i \quad \text{and} \quad \bigvee_{i \in I} \Phi_i$$

when I is finite.

(d) Similarly

$$(\exists z \leqslant x)\Phi_1,$$

$$(\forall z \leqslant x)\Phi_1$$

represent the results of bounded quantification.

(e) Although explicit transformations have already been allowed for, we have a further possibility for substitution. If $f : \mathbb{N}^m \to \mathbb{N}$ is such that

$$\left.\begin{array}{l} - \text{ the } (k+1) \text{ place relation } f(x_1,\ldots,x_k) = y \text{ is in } \mathscr{C} \text{ and} \\ - \text{ for some } i, f(x_1,\ldots,x_k) \leqslant x_i \end{array}\right\} \tag{5}$$

then ξ_i may make the form $f(x_1,\ldots,x_k)$. We can allow such substitutions because

$$\langle \ldots, f(x_1,\ldots,x_k), \ldots \rangle \in R$$

is equivalent to

$$(\exists z \leqslant x_i)(\langle \ldots, z, \ldots \rangle \in R \wedge f(x_1,\ldots,x_k) = z).$$

If $g_1,\ldots,g_k : \mathbb{N}^q \to \mathbb{N}$ have the same properties, then the substitution $f(g_1(x_1,\ldots,x_q), \ldots, g_k(x_1,\ldots,x_q))$ is also allowed, etc.

Examples. The relation $x < y$ is defined by the formula

$$(x \leqslant y) \wedge (x \neq y)$$

where $x \neq y$ abbreviates $\neg(x = y)$ and $x = y + 1$ defined by the formula

$$y \leqslant x \wedge \neg(\exists z \leqslant y)(y < z \wedge z < x).$$

Therefore both these relations are in any Δ_0-closed class. The function $\lambda x.x - 1 : \mathbb{N} \to \mathbb{N}$, where $x - 1 = y$ is the relation

$$y = 0 \vee (x = y + 1),$$

has the properties at (5).

For Δ_0-closed $\mathscr{C}$, we can also see that:

(a) If $F \in \mathscr{C}$ and X is any subset of G (therefore finite) then the formula

$$F(\xi_1,\ldots,\xi_m) \in X$$

represents a relation in $\mathscr{C}$, for it is equivalent to

$$\langle \xi_1,\ldots,\xi_m \rangle \in \bigvee_{\alpha \in X} F^{-1}(\alpha).$$

(b) If $F_1,\ldots,F_k,F_{k+1} \in \mathscr{C}$, $\Phi_1,\ldots,\Phi_k \in \mathscr{C}$ and

$$F(x_1,\ldots,x_m) = \begin{cases} F_1(\xi_1,\ldots,\xi_k), & \text{if } \Phi_1 \text{ else} \\ \vdots & \\ F_k(\xi_1,\ldots,\xi_k), & \text{if } \Phi_k \text{ else} \\ F_{k+1}(\xi_1,\ldots,\xi_k) & \end{cases}$$

then $F \in \mathscr{C}$. As before, the ξ_i may include certain function symbols. Thus, for instance, if $F_1 \in \mathscr{C}$ and

$$F(x) = F_1(x - 1)$$

then $F \in \mathscr{C}$.

(c) If $F_1,\ldots,F_k \in \mathscr{C}$, if $\psi: G^k \to G$ is any operation on the underlying set of G, and if

$$F(x_1,\ldots,x_m) = \psi(F_1(x_1,\ldots,x_m),\ldots,F_k(x_1,\ldots,x_m))$$

then $F \in \mathscr{C}$.

In particular ψ may be the semigroup operation, that is $\psi(\alpha,\beta) = \alpha \circledast \beta$, or if G is a group ψ may be the inverse operation,[4] that is $\psi(\alpha) = \alpha^{-1}$.

Immediate observations

Definition 12. Let $[k,i] = \{x \in \mathbb{N}: \ k \leqslant x \leqslant i\}$.

We can extend the definition of $\overline{F}(i)$ by defining

$$\overline{F}[k,i] = F(k) \circledast F(k+1) \circledast \cdots \circledast F(i),$$

so that $\overline{F}(i) = \overline{F}[0,i]$.

If G is a monoid (i.e. possesses an identity) and we have a Δ_0-closed class $\mathscr{C}$ of relations which is also closed under deterministic summation *mod* G then $F \in \mathscr{C}$ implies $\overline{F}[\cdot,\cdot] \in \mathscr{C}$. If we define $F_1 : \mathbb{N} \to G$ by

$$F_1(j,k) = \begin{cases} \underline{id} & \text{if } j < k, \\ F(j) & \text{if } j \geqslant k \end{cases}$$

(making the parameter k explicit) then, for all $k \leqslant i$,

$$\overline{F}[k,i] = \overline{F_1}(i,k).$$

We shall, in places, be relying implicitly on the fact that in this move to $\overline{F}[k,i]$, k is a parameter and not a constant.

As a convention, if $i < k$, $\overline{F}[k,i] = \underline{id}$.

Lemma 1. *If $\mathscr{L}$ is (isomorphic to) a subsemigroup of G then closure under deterministic summation mod G implies closure under deterministic summation mod $\mathscr{L}$.*

Proof. Trivial. □

3. The main theorem

The main result is as follows:

Theorem 1. *Assume Δ_0-closure. Then for all $n \in \mathbb{N}$, closure under determinstic summation mod S_n implies closure under deterministic summation mod $\mathscr{B}_n$.*

[4] Here and occasionally below, $^{-1}$ has a slightly different sense from that in Definition 3 but no confusion should arise.

Proof. The proof is an induction on n.

It is relatively straightforward to see that:

(i) Summation $mod\,\mathscr{B}_{n-1}$ implies summation $mod\,\mathscr{B}_n \backslash \mathscr{T}_n^\sharp$ The latter, recall, consists of those elements of $\mathscr{B}_n$ with domain of size $< n$ (Lemma 4).

(ii) Summation $mod\,\mathscr{B}_n \backslash \mathscr{T}_n^\sharp$ and summation $mod\,\mathscr{T}_n^\sharp$ together imply summation $mod\,\mathscr{B}_n$ (Lemma 7).

(iii) Summation $mod\,S_n$ implies summation $mod\,\mathscr{T}_n$ (Lemmas 5 and 6).

These obtained, there remains only the key step.

(iv) Summation $mod\,\mathscr{B}_{n-1}$ and summation $mod\,\mathscr{T}_n$ together imply summation $mod\,\mathscr{T}_n^\sharp$ (Lemma 9). This is elaborated below. □

Corollary 1.

$$\mathscr{B}_n\text{-}\Delta_0[\mathscr{F}] = S_n\text{-}\Delta_0[\mathscr{F}].$$

Outline of the argument

Assume summation $mod\,\mathscr{B}_{n-1}$ and summation $mod\,\mathscr{T}_n$. How to show summation $mod\,\mathscr{T}_n^\sharp$?

Suppose, for example, that we are trying to sum the sequence $F: \mathbb{N} \to \mathscr{T}_3^\sharp$ of Fig. 1. The sum we aim for is $\overline{F}: \mathbb{N} \to \mathscr{T}_3^\sharp$, is shown in Fig. 2.

Since we are summing summation $mod\,\mathscr{B}_{n-1}$, one avenue is to subdivide the domain, i.e. $[3] = \{0,1,2\}$ into (a) a singleton subdomain isomorphic to $[1] = \{0\}$ and (b) its complement, isomorphic to $[2] = \{0,1\}$, perhaps as in Fig. 3. This produces two sequences, one taking values in $\mathscr{B}_1$ and the other in $\mathscr{B}_2$, both of which we can sum. Unfortunately this ignores crossings from one subdivision to the other ($j \in \{0,2,3,4,5,\ldots\}$). Nonetheless the position is not completely hopeless: If all the crossings are one-way, say from top to bottom, then once we cross over, we cannot back. In effect each crossing splits the problem into two parts: summation in the top domain before the crossing, and summation in the bottom domain after the crossing.

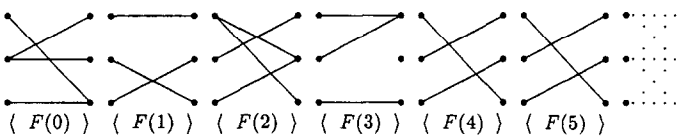


Fig. 1. $F: \mathbb{N} \to \mathscr{T}_3^\sharp$.

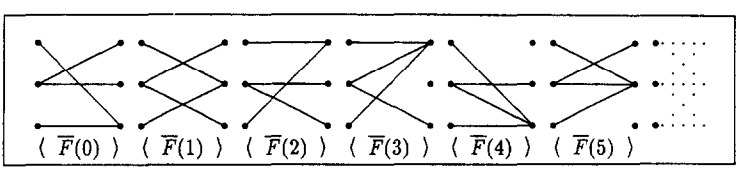


Fig. 2. $\overline{F}: \mathbb{N} \to \mathscr{T}_3^\sharp$.

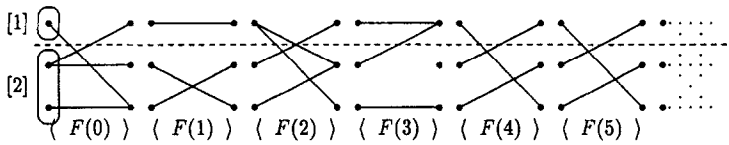


Fig. 3. Possible subdivision.

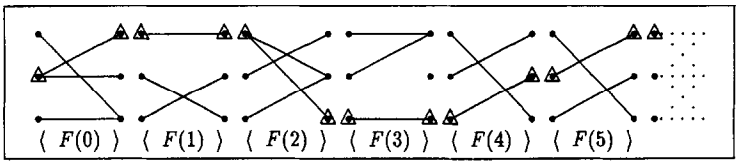
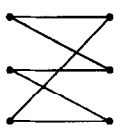


Fig. 4. Possible moving subdivisions with the "one-way" property.

These are summable by assumption, and, moreover, bounded existential quantification would allow us to try all possible crossing points.

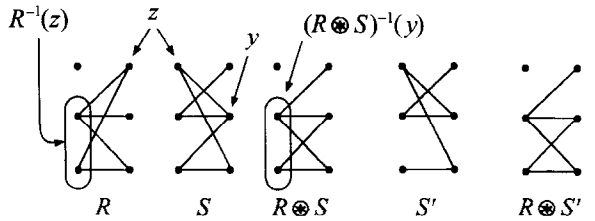
Unfortunately, our example sequence and subdivisions do not possess this "one-way" property (at $F(3)$ the only crossing is top-to-bottom, but for $j \in \{0,2,4,5\}$ there are crossings both ways). However, there is something more we can do, because the subdivisions need not be fixed. We can use a different subdivision at each boundary between $F(j)$ and $F(j+1)$. In Fig. 4, we see such a possibility, with the singleton subdivisions picked out by $\triangle$'s. These moving subdivisions have the "one-way" property.

Alas, there are two major problems. Firstly, though these subdivisions work, we have not defined them using only Δ_0-operations and the simpler summations of the induction hypothesis. Indeed, at the level of definability, finding a sequence of subdivisions with the one-way property is as hard as doing the original summation. Secondly, subdivision may be impossible. For example, the multifunction



does not permit it.

We would like to find a general method of modifying a sequence of elements of $\mathscr{T}_n^\sharp$ so as to achieve the "one-way" property. The method we adopt is to fix the subdivisions first, and then remove any crossings which go the wrong way. The hope is that we will not be removing any useful information. The reason for believing that this might be possible is as follows. Take a composite $R \circledast S$. If $\langle z, y\rangle \in S$ then, certainly, $R^{-1}(z) \subseteq (R \circledast S)^{-1}(y)$. Logically, therefore, either $R^{-1}(z) = (R \circledast S)^{-1}(y)$ or $R^{-1}(z) \subsetneq (R \circledast S)^{-1}(y)$. Suppose that the former holds. In this case any pair $\langle w, y\rangle \in S$ with $w \neq z$ is redundant; we can remove such pairs, producing an S' such that $R \circledast S' = R \circledast S$. For example:



Ignore, for the moment, the case $R^{-1}(z) \subsetneq (R \circledast S)^{-1}(y)$ and return to our sequence F. We need to be more precise about how the subdivisions are fixed, defining them using only Δ_0-operations and the simpler summations. It is here that we use summation $mod \mathscr{T}_n$. Fig. 5 shows a sequence $F_1 : \mathbb{N} \to \mathscr{T}_3$. Each $F_1(j)$ is a subrelation of $F(j)$ and it is possible to extract F_1 from F using only boolean operations. (F_1 and F_3 (below) are the same as in the proof of Lemma 9). Because, by assumption, we can sum F_1 to give $\overline{F_1}$, we immediately have access to the sequence $P : \mathbb{N} \to [3]$, marked by ◯'s, which defines the singletons in our sequence of subdivisions.

Having fixed the subdivisions, Δ_0-operations suffice to remove crossings going the wrong way. In our example, this is necessary for $F(2)$ and $F(3)$, both of which lose a crossing. The result is $F_3 : \mathbb{N} \to \mathscr{B}_3$, shown in Fig. 6. Fig. 7 shows more clearly the subdivisions and the fact that the crossings are one-way. Fig. 8 shows how a crossing splits the problem into two simpler summations, one before the crossing and one after.

Lemma 8 will show that Δ_0-operations and the simpler summations yield $\overline{F_3} : \mathbb{N} \to \mathscr{B}_3$ (Figs. 9 and 10).

We obtain F_3 by removing information from F; therefore, for all $j \in \mathbb{N}$, $F_3(j) \subseteq F(j)$, and hence $\overline{F_3}(j) (\subseteq \overline{F}(j)$. What we might hope is that, in fact, $\overline{F_3}(j) = \overline{F}(j)$, and indeed this is true for $0 \leqslant j \leqslant 2$. What goes wrong at $j = 3$?

The answer is that, for $0 < j \leqslant 2$,

$$\overline{F}(j-1)(P(j)) = \overline{F}(j)P(j+1)) = (\overline{F}(j-1) \circledast F(j))(P(j+1))$$

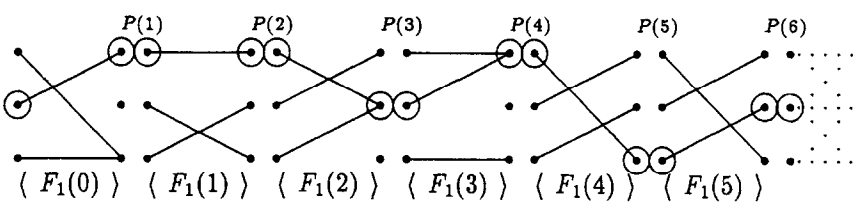


Fig. 5. $F_1 : \mathbb{N} \to T_3, P : \mathbb{N} \to [3]$

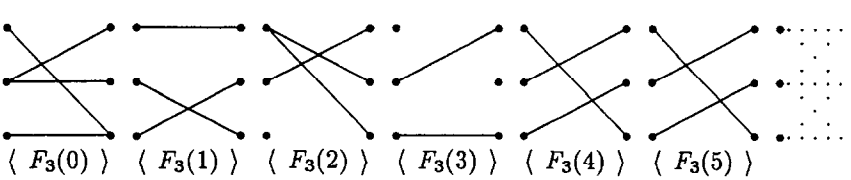


Fig. 6. $F_3 : \mathbb{N} \to B_3$

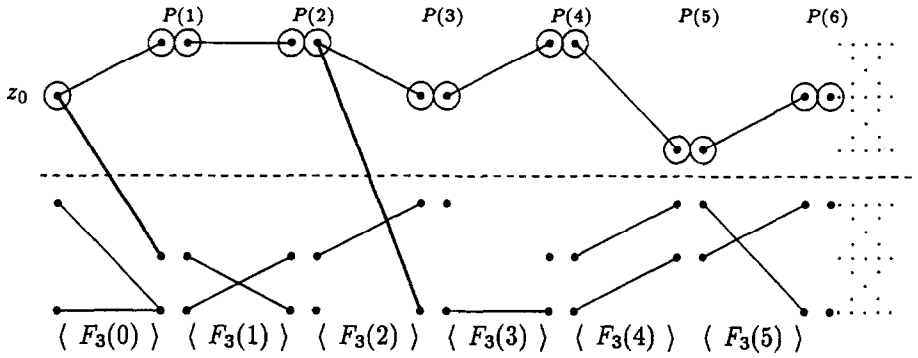


Fig. 7. $F_3 : \mathbb{N} \to B_3$, spread.

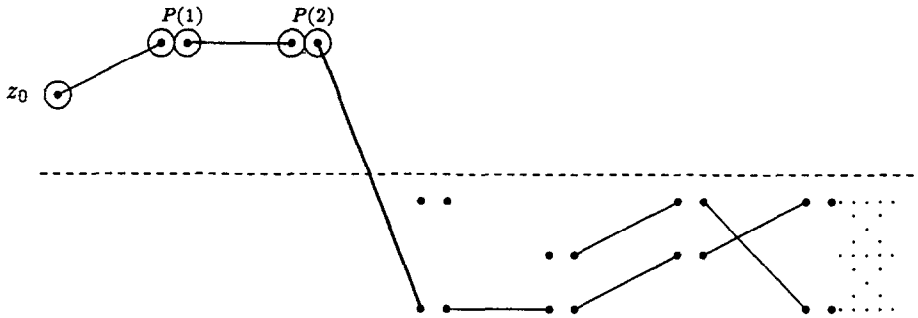


Fig. 8. A crossing.

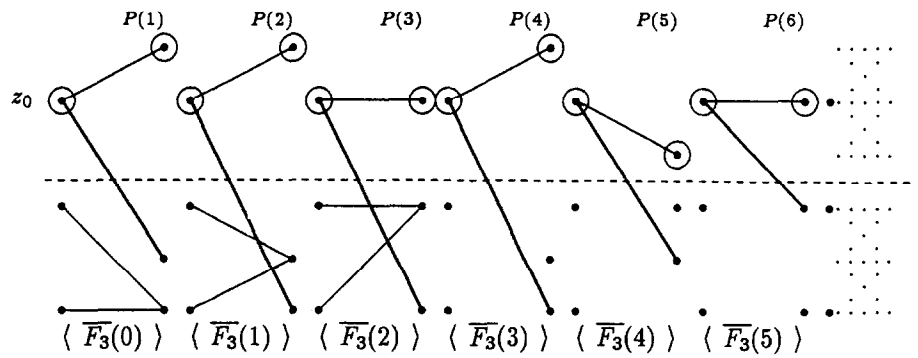


Fig. 9. $\bar{F} : \mathbb{N} \to B_3$, spread.

and therefore, as argued above, any pair $\langle w, P(j+1)\rangle \in F(j)$, with $w \neq P(j)$, can be removed (from $F(j)$). This is precisely what we did to obtain $F_3(j)$.

But at $j = 3$ (see Fig. 11), we encounter the other case,

$$\overline{F}(j-1)(P(j)) \subsetneq \overline{F}(j)(P(j+1))$$

and the argument breaks down. What can we do to rescue it?

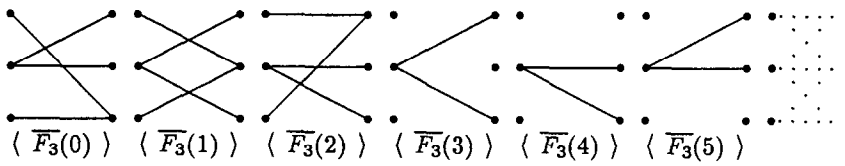


Fig. 10. $\bar{F} : \mathbb{N} \to B_3$

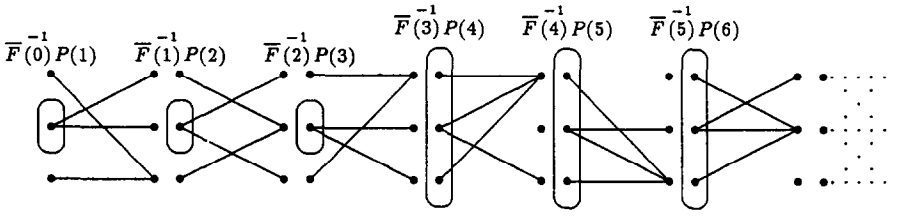


Fig. 11. $\bar{F}(j)^{-1}(P(j+1)) : \mathbb{N} \to \wp([3])$.

The solution is to observe that there is a finite bound on the number of times that

$$\overline{F}(j-1)(P(j)) \subsetneq \overline{F}(j)(P(j+1)).$$

Indeed this can happen at most n times, for what we have is a nondecreasing chain of subsets of $[n]$.

In our example it only happens once, at $j = 3$, since

$$\overline{F}(3)(P(4)) = [3]$$

is as big as possible. We can therefore split the sequence $F : \mathbb{N} \to \mathscr{T}_3^\sharp$ into two blocks separated by $F(3)$. For $j \leqslant 2$, we know that $\overline{F}(j) = \overline{F}_3(j)$, because the information lost in the move to F_3 is redundant. At $j = 3$ there is a problem but, for $j > 3$, information lost is again redundant; for $j > 3$, we can assert

$$\overline{F}(i) = \overline{F}(3) \circledast F_3(4) \circledast \cdots \circledast F_3(i) = \overline{F}(3) \circledast \overline{F_3}[4, i],$$

recalling the notation introduced in Definition 12. As we remarked following that definition, the sum $\overline{F_3}[4, i]$ is almost as easily defined as $\overline{F_3}(i)$. $\overline{F}(3) = \overline{F}(2) \circledast F(3)$ is available too, of course. This gives us the sum $\overline{F}(i)$, for all $i \in \mathbb{N}$ (Figs. 12 and 13). We are almost done.

One snag remains: finding the points where

$$\overline{F}(j-1)(P(j)) \subsetneq \overline{F}(j)(P(j+1))$$

is just as hard as the original problem. Fortunately, we do not need to find them! Just as we could use one bounded existential quantifier to try all possible crossings, we can use $n-1$ bounded existential quantifiers to try all possible combinations of points at which

$$\overline{F}(j-1)(P(j)) \subsetneq \overline{F}(j)(P(j+1)).$$

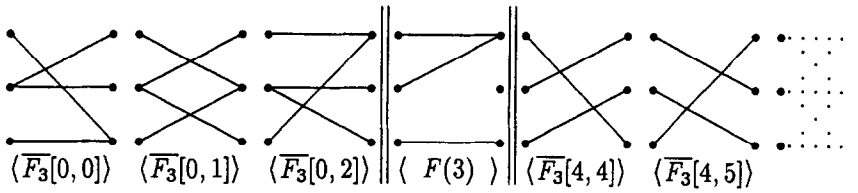


Fig. 12. Split about $F(3)$.

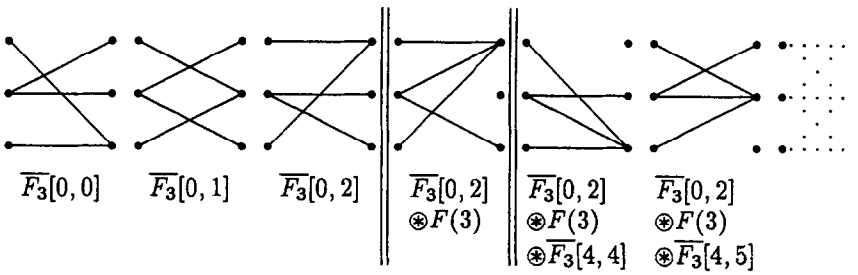


Fig. 13. $\overline{F}$ ragained.

"Wrong" tries do no damage because we never add information, only remove it. What matters is that, for at least one try (a single split about $F(3)$ in our example above), the information removed turns out to be redundant, in which case we regain the whole of $\overline{F}$.

This concludes the argument. The proofs of Lemmas 8 and 9 give full details.

4. Two technical lemmas

This section provides two technical lemmas which make the later work easier. The following is essentially [9, Lemma IV.3].

Lemma 2. *Suppose that $\mathscr{C}$ is Δ_0-closed. Let J be a monoid and H a semigroup, both contained in some larger finite monoid G, which they generate. Suppose that $\mathscr{C}$ is closed under deterministic summation mod J and mod H. Suppose that for all $\beta \in J$, $\gamma \in H$, there is a $\gamma \boxtimes \beta \in H$ such that*

$$\gamma \circledast \beta = \gamma \circledast (\gamma \boxtimes \beta).$$

Then $\mathscr{C}$ is closed under deterministic summation mod G.

Proof. This lemma might at first seem trivially true, but being able to compose sequences of elements of J and (separately) sequences of elements of H does not imply directly that we can compose mixed sequences.

Let

$$F : \mathbb{N} \to G$$

be in $\mathscr{C}$.

From the assumed property of $\boxtimes$ it follows that

$$H \circledast J = \{\gamma \circledast \beta\colon\ \gamma \in H \text{ and } \beta \in J\} \subseteq H$$

and hence

$$G = J \cup H \cup (J \circledast H).$$

Also, there must be operations $\beta\colon G \to J$ and $\gamma\colon G \to H$ such that, for all $\alpha \in J \circledast H$,

$$\alpha = \beta(\alpha) \circledast \gamma(\alpha)$$

with β and γ defined arbitrarily elsewhere, i.e. on $(J \cup H)\backslash(J \circledast H)$. (There may be more than one choice for $\beta(\alpha)$ and $\gamma(\alpha)$: we fix on one.)

1. The first step is a reduction to a function F_2 with $F_2(0) \notin J$. Let $F_1 : \mathbb{N} \to J$ be defined by

$$F_1(j) = \begin{cases} F(j) & \text{if } F(j) \in J, \\ \underline{id}_J & \text{if } F(j) \notin J. \end{cases}$$

Since J is a monoid, $F_1[\cdot,\cdot] \in \mathscr{C}$.

Let $p\colon \mathbb{N} \to \mathbb{N}$ be defined by $p(w) = y$ if and only if

$$\begin{array}{llll} & y = 0 & \wedge & F(0) \notin J \\ \vee & y > 0 & \wedge & F(y) \notin J \wedge (\forall i \leqslant y-1)(F(i) \in J) \\ \vee & y = 0 & \wedge & (\forall i \leqslant w)(F(i) \in J) \end{array}$$

(so $p(w)$ is the least $y \leqslant w$ such that $F_1(y) \notin J$, if such exists, and is 0 otherwise). Clearly $p(w) \leqslant w$ and p satisfies the substitution conditions (5) above.

Let

$$F_2(j) = \begin{cases} F(0) & \text{if } p(w) = 0 \wedge j = 0, \\ \overline{F_1}[0, p(w)-1] \circledast F(p(w)) & \text{if } p(w) > 0 \wedge j = 0, \\ \underline{id}_G & \text{if } p(w) > 0 \wedge j > 0 \wedge j \leqslant p(w), \\ F(j) & \text{if } j > p(w). \end{cases}$$

(F_2 has w as a parameter.) It is easy to see that $\overline{F_2}(p(w)) = \overline{F}(p(w))$ and hence, for all $p(w) \leqslant j \leqslant w$,

$$\overline{F_2}(j) = \overline{F}(j).$$

Thus

$$\overline{F}(w) = \begin{cases} \overline{F_1}(w) & \text{if } (\forall i \leqslant w)(F(i) \in J), \\ \overline{F_2}(w) & \text{if } (\exists i \leqslant w)(F(i) \notin J). \end{cases}$$

Therefore $\overline{F_2} \in \mathscr{C}$ would imply $\overline{F} \in \mathscr{C}$.

2. The next step is to eliminate all values in J from F_2. Let $\ell : \mathbb{N} \to \mathbb{N}$ be defined by $\ell(j) = y$ if and only if

$$y \leqslant j \wedge F_2(y-1) \notin J \wedge (\forall i \leqslant j)(i < y \wedge F_2(i) \in J)$$

(so that $\ell(j)-1$ is the greatest $z \leqslant y$ such that $F_2(z) \notin J$). Clearly $\ell(j) \leqslant j$ and ℓ satisfies the substitution conditions (5) above.

Let

$$F_3(j)=\begin{cases} F_2(j) & \text{if } F_2(j) \notin J, \\ F_2(\ell(j)-1)\boxtimes \overline{F}_1[\ell(j),j] & \text{if } F_2(j) \in J \wedge F_2(\ell(j)-1) \in H, \\ (\gamma(F_2(\ell(j)-1)))\boxtimes \overline{F}_1[\ell(j),j] & \text{if } F_2(j) \in J \wedge F_2(\ell(j)-1) \in (J \circledast H)\backslash H \end{cases}$$

where $\gamma : G \to H$ is the operation fixed on above.
$F_3 \in \mathscr{C}$ because $\overline{F}_1[\ell(j),j]$ is a composite of elements of the monoid J. An easy induction shows that for all j with $F_2(j) \in J$,

$$\overline{F}_3[\ell(j)-1,j] = \overline{F}_2[\ell(j)-1,j].$$

Hence $\overline{F}_2(j) = \overline{F}_3(j)$ and, for all $j \in \mathbb{N}$, $F_3(j) \notin J$.

For example, if $F_2(7) \in (J \circledast H)\backslash H$ and $F_2(8), F_2(9) \in J$ then $\ell(9) = 8$ and

$$\begin{aligned} \overline{F}_2[7,9] &= \beta(F_2(7)) \circledast \gamma(F_2(7)) \circledast F_2(8) \circledast F_2(9) \\ &= \beta(F_2(7)) \circledast \gamma(F_2(7)) \circledast F_2(8) \\ &\quad \circledast((\gamma(F_2(7)) \circledast F_2(8))\boxtimes F_2(9)) \\ &= \beta(F_2(7)) \circledast \gamma(F_2(7)) \circledast (\gamma(F_2(7))\boxtimes F_2(8)) \\ &\quad \circledast((\gamma(F_2(7)) \circledast F_2(8))\boxtimes F_2(9)) \\ &= F_3(7) \circledast F_3(8) \circledast F_3(9) = \overline{F}_3[7,9] \end{aligned}$$

3. Now work on F_3. The third step is to eliminate all values in $(J \circledast H)\backslash H$ from F_3, reducing it to a function F_4 which takes values only in H. Again we use the operations $\beta : G \to J$ and $\gamma : G \to H$.

Define $F_4(j)$ by

$$F_4(j) = \begin{cases} F_3(j) & \text{if } F_3(j) \in H, \\ \gamma(F_3(0)) & \text{if } F_3(0) \notin H \wedge j = 0, \\ (F_3(j-1)\boxtimes \beta(F_3(j))) \circledast \gamma(F_3(j)) & \\ \qquad \text{if } F_3(j) \notin H \wedge j > 0 \wedge F_3(j-1) \in H, & \\ (\gamma(F_3(j-1))\boxtimes \beta(F_3(j))) \circledast \gamma(F_3(j)) & \\ \qquad \text{if } F_3(j) \notin H \wedge j > 0 \wedge F_3(j-1) \notin H. & \end{cases}$$

Then, for all $j \in \mathbb{N}$,

$$\overline{F}_3(j) = \begin{cases} \overline{F}_4(j) & \text{if } F_3(0) \in H, \\ \beta(F_3(0)) \circledast \overline{F}_4(j) & \text{if } F_3(0) \notin H. \end{cases}$$

Thus $\overline{F}_4 \in \mathscr{C}$ would imply $\overline{F}_3 \in \mathscr{C}$.

But $F_4 \in \mathscr{C}$ and $F_4 : \mathbb{N} \to H$. Hence $\overline{F}_4 \in \mathscr{C}$, and we deduce $\overline{F} \in \mathscr{C}$. □

Lemma 3. *Suppose that $\mathscr{C}$ is Δ_0-closed. Let J be a monoid and H a semigroup, both contained in some larger (finite) monoid G, which they generate. Suppose that $\mathscr{C}$ is closed under deterministic summation mod J and mod H. Suppose that:*

(i) $H \circledast J \subseteq H$;

(ii) *for all $\gamma \in H$, there is an $\iota^{\gamma} \in H$ such that*

$$\iota^{\gamma} \circledast \gamma = \gamma.$$

It follows that $\mathscr{C}$ is closed under deterministic summation mod G.

Proof. As in proof of Lemma 2, $G = J \cup H \cup (J \circledast H)$ and we can define operations $\beta : G \to J$ and $\gamma : G \to H$ such that for each $\alpha \notin H \cup J$

$$\alpha = \beta(\alpha) \circledast \gamma(\alpha)$$

with β and γ defined arbitrarily on $G \backslash (H \cup J)$.

If

$$F : \mathbb{N} \to G$$

define

$$F_1(j) = \begin{cases} F(j) & \text{if } F(j) \in H \cup J \wedge [(j = 0) \vee (F(j-1) \in H \cup J)], \\ \gamma(F(j-1)) \circledast F(j) & \text{if } F(j) \in H \cup J \wedge [(j > 0) \vee (F(j-1) \notin H \cup J)], \\ \gamma(F(j-1)) \circledast \beta(F(j)) & \text{if } F(j) \notin H \cup J \wedge [(j > 0) \vee (F(j-1) \notin H \cup J)], \\ \beta(F(j)) & \text{if } F(j) \notin H \cup J \wedge [(j = 0) \vee (F(j-1) \in H \cup J)], \end{cases}$$

then $F_1 : \mathbb{N} - H \cup J$ and an easy induction shows that

$$\overline{F}(w) = \begin{cases} \overline{F}_1(w) \circledast \gamma(F(w)) & \text{if } F(w) \notin H \cup J, \\ \overline{F}_1(w) & \text{if } F(w) \in H \cup J. \end{cases}$$

Let $F_2 : \mathbb{N} \to J$ be defined by

$$F_2(j) = \begin{cases} F_1(j) & \text{if } F_1(j) \in J, \\ \underline{id}_J & \text{if } F_1(j) \notin J. \end{cases}$$

With $p : \mathbb{N} \to \mathbb{N}$ defined by $p(w) = y$ if and only if

$$\begin{aligned} & [y = 0 \ \wedge \ F_1(0) \notin J] \\ \vee \ & [y > 0 \ \wedge \ F_1(y) \notin J \wedge (\forall i \leqslant y - 1)(F_1(i) \in J)] \\ \vee \ & [y = 0 \ \wedge \ (\forall i \leqslant w)(F_1(i) \in J)] \end{aligned}$$

(so $p(w)$ is the least $y \leqslant w$ such that $F_1(y) \notin J$, if such exists, and is 0 otherwise), let

$$F_3(j) = \begin{cases} \iota^{F_1(p(w))} & \text{if } j < p(w), \\ F_1(j) & \text{if } j \geqslant p(w) \end{cases}$$

so that $F_3(0) \in H$. It is easy to see that

$$\overline{F}_1(w) = \begin{cases} \overline{F}_2(p(w)-1) \circledast \overline{F}_3(w) & \text{if } p(w) > 0, \\ \overline{F}_2(p(w)), & \text{if } (\forall i \leqslant w)(F_1(i) \in J). \end{cases}$$

Let $q : \mathbb{N}^2 \to \mathbb{N}$ be defined by $q(w,j) = y$ if and only if

$$\begin{aligned} & j < y \wedge y \leqslant w \wedge F_3(y) \in H \ \wedge \ (\forall i \leqslant y-1)(i \leqslant j \vee F_3(i) \notin H) \\ \vee \ & y = 0 \qquad\qquad\qquad\qquad \wedge \ (\forall i \leqslant w)(i \leqslant j \vee F_3(i) \notin H) \end{aligned}$$

(so that $q(w,j)$ is the least $j < y \leqslant w$ such that $F_3(y) \in H$, if such exists, and is 0 otherwise). Let

$$F_4(j) = \begin{cases} \iota^{F_3(0)} & \text{if } j = 0, \\ F_3(j-1) & \text{if } j > 0 \ \wedge \ F_3(j-1) \in H \wedge F_3(j) \in H, \\ F_3(j-1) \circledast \overline{F}_2[j, q(w,j)-1] & \text{if } j > 0 \ \wedge \ F_3(j-1) \in H \wedge F_3(j) \notin H, \\ \iota^{F_3(q(w,j))} & \text{if } j > 0 \ \wedge \ F_3(j-1) \notin H. \end{cases}$$

Then with $r : \mathbb{N} \to \mathbb{N}$ defined by $r(w) = y$ if and only if

$$y \leqslant w \wedge F_3(y-1) \in H \wedge (\forall i \leqslant w)(i < y \vee F_3(i) \notin H)$$

(so that $F_3(w) \notin H$ implies that $(r(w)-1)$ is the largest $z < w$ such that $F_3(y) \in H$), we have

$$\overline{F}_3(w) = \begin{cases} \overline{F}_4(w) \circledast F_3(w) & \text{if } F_3(w) \in H, \\ \overline{F}_4(r(w)) \circledast \overline{F}_2[r(w), w] & \text{if } F_3(w) \notin H. \end{cases}$$

But $F_4 : \mathbb{N} \to H$ and $F_4 \in \mathscr{C}$. Therefore $\overline{F}_4 \in \mathscr{C}$. □

5. Four easy lemmas

The lemmas in this section are more specific to the application.

Lemma 4. *For $n \geqslant 1$, deterministic summation mod $\mathscr{B}_{n-1}$ implies deterministic summation mod $\mathscr{B}_n \backslash \mathscr{T}_n^\sharp$. (assuming Δ_0-closure).*

Proof. Let the operation $\delta : (\mathscr{B}_n \backslash \mathscr{T}_n^\sharp) \to ([n-1] \to [n])$ be such that, for all $\rho \in \mathscr{B}_n \backslash \mathscr{T}_n^\sharp$,

$$\delta(p) : [n-1] \overset{1\text{–}1}{\to} [n] \quad \text{and} \quad \mathrm{Dom}(\rho) \subseteq \delta(\rho).$$

This is possible because $\rho \in \mathscr{B}_n \backslash \mathscr{T}_n^\sharp$ implies $|\mathrm{Dom}(\rho)| < n$. Clearly

$$\rho = (\delta(\rho))^{-1} \circledast \delta(\rho) \circledast \rho.$$

Define $F_1 \in \mathscr{C}$ by

$$F_1(j) = \begin{cases} \delta(F(0)) \circledast (\delta(F(0)))^{-1} & \text{if } j = 0, \\ \delta(F(j-1)) \circledast F(j-1) \circledast (\delta(F(j)))^{-1} & \text{if } j > 0. \end{cases}$$

Then

$$\overline{F}(j) = (\delta(F(0)))^{-1} \circledast \overline{F}_1(j) \circledast \delta(F(j)) \circledast F(j).$$

But $F_1 : \mathbb{N} \to \mathscr{B}_{n-1}$ and hence, by assumption, $\overline{F}_1 \in \mathscr{C}$. Therefore $\overline{F} \in \mathscr{C}$. □

Lemma 5. *For $n \geqslant 1$, deterministic summation mod $\mathscr{T}_{n-1}$ implies deterministic summation mod $\mathscr{T}_n \backslash S_n$ (assuming Δ_0-closure).*

Proof. Let the operation $\zeta : (\mathscr{T}_n \backslash S_n) \to ([n-1] \to [n])$ be such that, for all $f \in \mathscr{T}_n \backslash S_n$,

$$\zeta(f) : [n-1] \overset{1\text{–}1}{\to} [n] \quad \text{and} \quad \mathrm{Ran}(f) \subseteq \zeta(\rho).$$

This is possible because $f \in \mathscr{T}_n \backslash S_n$ implies $|\mathrm{Ran}(f)| < n$. Clearly

$$f = f \circledast (\zeta(f))^{-1} \circledast \zeta(f).$$

Define $F_1 \in \mathscr{C}$ by

$$F_1(j) = \begin{cases} \zeta(F(0)) \circledast (\zeta(F(0)))^{-1} & \text{if } j = 0, \\ \zeta(F(j-1)) \circledast F(j) \circledast (\zeta(F(j)))^{-1} & \text{if } j > 0. \end{cases}$$

Then

$$\overline{F}(j) = F(0) \circledast (\zeta(F(0)))^{-1} \circledast \overline{F}_1(j) \circledast \zeta(F(j)).$$

But $F_1 : \mathbb{N} \to \mathscr{T}_{n-1}$ and hence, by assumption, $\overline{F}_1 \in \mathscr{C}$. Therefore $\overline{F} \in \mathscr{C}$. □

The remaining two lemmas in this section rely on those in Section 4.

Lemma 6. *Given Δ_0-closure, closure under deterministic summation mod S_n and mod $\mathscr{T}_n / S_n$ together imply closure under deterministic summation mod $\mathscr{T}_n$.*

Proof. This is a simple application of Lemma 2. Let $J = S_n$, the group of permutations on $[n]$, $n \geqslant 2$. Let $H = \mathscr{T}_n \backslash S_n$. Clearly both are closed under $\circledast$.

If $f \in \mathscr{T}_n \backslash S_n$ then f is not onto, for the only onto functions are permutations. So there is a least $a < n$ such that $a \notin \mathrm{Ran}(f)$.

But now if $\sigma \in S_n$, define $f \boxtimes \sigma$ by

$$(f \boxtimes \sigma)(x) = \begin{cases} \sigma(x) & \text{if } x \neq a, \\ \sigma(b) & \text{if } x = a, \end{cases}$$

where $b < n$ is least such that $b \neq a$.

$f \boxtimes \sigma$ is still a function, but it is no longer a permutation. Therefore $f \boxtimes \sigma \in \mathscr{T}_n \backslash S_n$. On the other hand, it is quite clear that

$$f \circledast \sigma = f \circledast (f \boxtimes \sigma).$$

Together S_n and $\mathscr{T}_n \backslash S_n$ generate the monoid $\mathscr{T}_n$ of functions with domain and co-domain $[n]$. (Indeed the latter is just their union.) Thus the conditions for Lemma 2 are satisfied. □

Lemma 7. *Closure under deterministic summation mod $\mathscr{T}_n^\sharp$ and deterministic summation mod $\mathscr{B}_n \backslash \mathscr{T}_n^\sharp$ together imply (modulo Δ_0-closure) deterministic summation mod $\mathscr{B}_n$.*

Proof. This is a direct application of Lemma 3.

Let $J = \mathscr{T}_n^\sharp$. Recall that $\alpha \in \mathscr{T}_n^\sharp$ if and only if α is a binary relation over n and $\mathrm{Dom}(\alpha) = [n]$. Let $H = \mathscr{B}_n \backslash \mathscr{T}_n^\sharp$, so that $\rho \in \mathscr{B}_n$ if and only if $|\mathrm{Dom}(\rho)| < n$. Clearly H and J generate $\mathscr{B}_n$ and:

(i) For all $\gamma \in H$ and $\beta \in J$, $(\gamma \circledast \beta)$ has the same domain as γ and is therefore in H.

(ii) If, for $\gamma \in H$, we set

$$\iota^\gamma = \underline{id}_{\mathrm{Dom}(\gamma)} = \{\langle x, x\rangle : \ x \in \mathrm{Dom}(\gamma)\}$$

then clearly

$$\iota^\gamma \circledast \gamma = \gamma.$$

Thus all the conditions for Lemma 3 are satisfied. □

6. Key lemmas

The main business of this section is Lemma 9. We begin however with:

Lemma 8. *Let $n \in \mathbb{N}$ Suppose that $\mathscr{C}$, a Δ_0-closed class of relations on $\mathbb{N}$, is also closed under deterministic summation mod $\mathscr{B}_{n-1}$. Suppose that $F : \mathbb{N} \to \mathscr{B}_n$ and $F \in \mathscr{C}$. Suppose that $\mathscr{C}$ also contains a sequence $P : \mathbb{N} \to [n]$ such that, for all $j \in \mathbb{N}$,*

$$F(j)^{-1}(P(j+1)) \subseteq \{P(j)\}.$$

It follows that $\overline{F} \in \mathscr{C}$.

Proof. For $i \in \mathbb{N}$ and $x, y \in [n]$, $\langle x, y\rangle \in \overline{F}(i)$ if and only if there exists a sequence

$$x = u_0, u_1, \ldots, u_i, u_{i+1} = y$$

such that, for all $0 \leqslant m \leqslant i$,

$$\langle u_m, u_{m+1}\rangle \in F(m).$$

Since, by assumption, $u_{m+1} = P(m+1)$ implies $u_m = P(m)$, an easy induction shows that

$$u_m \neq P(m) \text{ implies } u_k \neq P(k) \quad \text{for all } m \leqslant k \leqslant i+1.$$

There are therefore only three cases in which $\langle x, y\rangle \in \overline{F}(i)$:

1. For all $0 \leqslant m \leqslant i+1$, $u_m = P(m)$. In this case $x = P(0)$ and $y \in P(i+1)$.
2. For all $0 \leqslant m \leqslant i+1$, $u_m \neq P(m)$.
3. $x = P(0)$ and for some $i \geqslant 1$, $\langle P(k-1), u_k\rangle \in F(k-1)$ and for all $k \leqslant m \leqslant i+1$, $u_m \neq P(m)$.

(It is clearly impossible, for instance, that $x \neq P(0)$ and $y = P(i+1)$.)

Let

$$F_1(i) = \begin{cases} \{\langle P(0), P(i)\rangle\} & \text{if } (\forall j \leqslant i)(j = 0 \ \vee \ \langle P(j-1), P(j)\rangle \in F(j-1)), \\ \emptyset & \text{if } (\exists j \leqslant i)(j > 0 \ \wedge \ \langle P(j-1), P(j)\rangle \notin F(j-1)). \end{cases}$$

$P \in \mathscr{C}$ implies $F_1 \in \mathscr{C}$. Furthermore, for $i \geqslant 1$ and $y = P(i)$,

$$\langle x, y\rangle \in \overline{F}(i-1) \text{ iff } \langle x, y\rangle \in \overline{F}_1(i).$$

This covers case 1 above.

Let the operation $\delta : [n] \to ([n-1] \overset{1\text{–}1}{\to} [n])$ be such that

$$\mathrm{Ran}(\delta(m)) = [n] \backslash \{m\}$$

where $([n-1] \overset{1\text{–}1}{\to} [n])$ is the set of 1–1 functions with domain $[n-1]$ and codomain $[n]$. Such operations clearly exist. Let

$$F_2(j) = \delta(P(j))^{-1} \circledast \delta(P(j)) \circledast F(j).$$

Observe that, if $u \neq P(j)$,

$$\langle u, v\rangle \in F(j) \text{ iff } \langle u, v\rangle \in F_2(j).$$

Clearly $F_2(j) \subseteq F(j)$, $j \in \mathbb{N}$, and so $\overline{F}_2(i) \subseteq \overline{F}(i)$, for all $i \in \mathbb{N}$. Indeed, following case 2 above, we see that, for $x \neq P(0)$,

$$\langle x, y\rangle \in \overline{F}(i) \text{ iff } \langle x, y\rangle \in \overline{F}_2(i).$$

Although it is clear that $F_2 \in \mathscr{C}$, we need to show that $\overline{F}_2 \in \mathscr{C}$. Let $F_3 \in \mathscr{C}$ be defined by

$$F_3(j) = \begin{cases} \delta(P(0)) \circledast \delta(P(0))^{-1} & \text{if } j = 0, \\ \delta(P(j-1)) \circledast F(j-1) \circledast \delta(P(j))^{-1} & \text{if } j > 0. \end{cases}$$

An easy induction shows that, for all $i \in \mathbb{N}$,

$$\overline{F}_2(i) = \delta(P(0))^{-1} \circledast \overline{F}_3(i) \circledast \delta(P(i)) \circledast F(i).$$

But $F_3 : \mathbb{N} \to \mathscr{B}_{n-1}$. Therefore, by assumption, $\overline{F}_3 \in \mathscr{C}$, and hence $\overline{F}_2 \in \mathscr{C}$. Since $\mathscr{B}_n$ is a monoid, this also entails $\overline{F}_2[\cdot, \cdot] \in \mathscr{C}$.

For case 3, we observe that if, for $v \in [n]$, we express $i \in (F_4)^{-1}(v)$ by the formula

$$\bigvee_{u\in[n]} (\exists k \leqslant i) \begin{pmatrix} k > 0 \\ \wedge \ \langle P(0), P(k-1)\rangle \in \overline{F}_1(k-1) \\ \wedge \ \langle P(k-1), u\rangle \in F(k-1) \\ \wedge \ \langle u, v\rangle \in \overline{F}_2[k,i] \end{pmatrix}$$

then $F_4 : \mathbb{N} \to [n]$ and $F_4 \in \mathscr{C}$. We can then define $F_5 \in \mathscr{C}$, $F_5 : \mathbb{N} \to \mathscr{B}_n$, by

$$F_5(i) = \{P(0)\} \times F_4(i) = \{\langle P(0), w\rangle : \ w \in F_4(i)\}.$$

For all $i \in \mathbb{N}$ and for $x = P(0)$ and $y \neq P(i)$,

$$\langle x, y\rangle \in \overline{F}(i) \text{ iff } \langle x, y\rangle \in F_5(i).$$

We can therefore assert that $\overline{F} \in \mathscr{C}$, because

$$\overline{F}(i) = \begin{cases} F(0) & \text{if } i = 0, \\ \overline{F}(i-1) \circledast F(i) & \text{if } i > 0 \end{cases}$$

and, for $i \geqslant 1$,

$$\overline{F}(i-1) = \overline{F}_2(i-1) \cup F_1(i) \cup F_5(i-1). \qquad \square$$

Lemma 9. *Suppose that $\mathscr{C}$, a Δ_0-closed class of relations, is closed under deterministic summation mod $\mathscr{T}_n$ and closed under deterministic summation mod $\mathscr{B}_{n-1}$.*

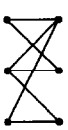
It follows that $\mathscr{C}$ is closed under deterministic summation mod $\mathscr{T}_n^\sharp$.

Proof. Suppose that $F \in \mathscr{C}$ and $F : \mathbb{N} \to \mathscr{T}_n^\sharp$. We shall first define a function $\Upsilon : \mathbb{N} \to \mathscr{T}_n^\sharp$, making it clear that $\Upsilon \in \mathscr{C}$. Then we shall show that in fact $\Upsilon = \overline{F}$.

Let $\theta : \mathscr{B}_n \to \mathscr{T}_n$ be an operation such that, for all $\rho \in \mathscr{T}_n^\sharp$,

$$\theta(\rho) \subseteq \rho$$

(that is to say $(\theta(\rho))(x) = y$ implies $\langle x, y\rangle \in \rho$, for all $x \in [n]$). For example, with $n = 3$,

for $\rho =$  , we could take $\theta(\rho) =$  .

If we define $F_1 : \mathbb{N} \to \mathscr{T}_n$ (recalling that $\mathscr{T}_n^\sharp \subseteq \mathscr{B}_n$) by

$$F_1(j) = \theta(F(j))$$

then $F_1 \in \mathscr{C}$. By assumption $\overline{F}_1 \in \mathscr{C}$; indeed $\overline{F}_1[\cdot,\cdot] \in \mathscr{C}$, because $\mathscr{T}_n$ is a monoid.

Now, for $\ell \leqslant n$, $z_0 \in [n]$ and $d_1, \ldots, d_\ell \in \mathbb{N}$, define $P_2 : \mathbb{N} \to [n]$ by

$$P_2(j) = \begin{cases} Z_0 & \text{if } j = 0, \\ (\overline{F}_1(j-1))(z_0) & \text{if } j > 0. \end{cases}$$

The form of the definition[5] ensures that $P_2 \in \mathscr{C}$. One may regard z_0 as a constant and $d_1, \ldots, d_\ell$ as parameters. Because $F_1(j) = \theta(F(j)) \subseteq F(j)$, for all $j \in \mathbb{N}$, it follows that

$$\langle P_2(j-1), P_2(j) \rangle \in F(j-1)$$

for all $j > 0$.

Define the operation $\tau : \mathscr{T}_n^\sharp \times [n] \times [n] \to \mathscr{B}_n$ by

$$\tau(\rho, y, z) = \{\langle w, v \rangle \in \rho : \ w = x \vee v \neq y\}.$$

If $\langle y, z \rangle \in \rho$ then

$$(\tau(\rho, y, z))^{-1}(z) = \{y\}.$$

Now define $F_3 : \mathbb{N} \to \mathscr{B}_n$, $F_3 \in \mathscr{C}$, by

$$F_3(j) = \begin{cases} \{\langle z_0, z_0 \rangle\} & \text{if } j = 0, \\ \tau(F(j-1), P_2(j-1), P_2(j)) & \text{if } j > 0. \end{cases}$$

By Lemma 8, taking $P(j) = P_2(j-1)$, $\overline{F}_3 \in \mathscr{C}$; also $\overline{F}_3[\cdot, \cdot] \in \mathscr{C}$, because $\mathscr{B}_n$ is a monoid.

With $d_1, \ldots, d_\ell$ as parameters define $F_4^\ell : \mathbb{N} \to \mathscr{B}_n$ by

$$F_4^\ell(i) = \begin{cases} \left.\begin{array}{l} \overline{F}_3[1, (d_1) - 1] \circledast F((d_1) - 1) \\ \circledast \overline{F}_3[(d_1) + 1, (d_2) - 1] \circledast F((d_2) - 1) \circledast \cdots \\ \cdots \circledast F((d_\ell) - 1) \circledast \overline{F}_3[(d_\ell) + 1, i] \circledast F(i) \end{array}\right\} & \text{if } d_\ell < i, \\ \left.\begin{array}{l} \overline{F}_3[1, (d_1) - 1] \circledast F((d_1) - 1) \\ \circledast \overline{F}_3[(d_1) + 1, (d_2) - 1] \circledast F((d_2) - 1) \circledast \cdots \\ \cdots \circledast F((d_\ell) - 1) \circledast F(i) \end{array}\right\} & \text{if } d_\ell = i. \end{cases}$$

This definition assumes that $d_1 < d_2 < \cdots d_\ell \leqslant i$, which is made explicit below. $F_4^\ell \in \mathscr{C}$, because the composites are finite and can be expressed as the results of operation $\mathscr{B}_n^{2\ell+2} \to \mathscr{B}_n$ and $\mathscr{B}_n^{2\ell+1} \to \mathscr{B}_n$. For the sake of well-definedness, recall our convention that, for $i < k$, $\overline{F}_3[k, i] = \underline{id}$. (The function expressed $(d_m) + 1$ should be thought of as having two arguments: d_m and i.)

[5] Proceeding with more caution we might have defined an operation App : $\mathscr{T}_n \times [n] \to [n]$ with $\mathrm{App}(f, x) = f(x)$, and written

$$\mathrm{App}(\overline{F}_1(j-1), z_0)$$

instead of $(\overline{F}_1(j-1))(z_0)$.

Since $\tau(\rho,y,z)\subseteq\rho$, for any $\rho\in\mathscr{T}_n^\sharp$, it is clear that $F_3(j+1)\subseteq F(j)$, for all $j\in\mathbb{N}$, and hence that $F_4^\ell(i)\subseteq\overline{F}(i)$, for all $i\geqslant d_\ell$.

Now define $\Gamma^\ell:\mathbb{N}\to\mathscr{B}_n$ by having $i\in(\Gamma^\ell)^{-1}(\rho)$ if and only if

$$\begin{aligned}&(\forall d_1\leqslant i)(\forall d_2\leqslant i)\cdots(\forall d_\ell\leqslant i)\\&\quad(d_1<1\vee d_2\leqslant d_1\vee\cdots\vee d_\ell\leqslant d_{\ell-1}\vee F_4^\ell(i)\subseteq\rho)\\&\qquad\qquad\wedge\\&(\exists d_1\leqslant i)(\exists d_2\leqslant i)\cdots(\exists d_\ell\leqslant i)\\&\quad(1\leqslant d_1\wedge d_1<d_2\wedge\cdots\wedge d_{\ell-1}<d_\ell\wedge F_4^\ell(i)=\rho).\end{aligned}$$

Making the parameters explicit, $\Gamma^\ell(i)$ is the union of $F_4^\ell(i,d_1,\ldots,d_\ell)$ over all appropriate sequences $d_1,\ldots,d_\ell$. Clearly $\Gamma^\ell\in\mathscr{C}$ and $\Gamma^\ell(i)\subseteq\overline{F}(i)$, for all $i\geqslant\ell$.

Finally define $\Upsilon:\mathbb{N}\to\mathscr{B}$ by having $i\in\Upsilon^{-1}(\rho)$ if and only if

$$\bigwedge_{\ell\leqslant n}(i<\ell\vee\Gamma^\ell(i)\subseteq\rho)\wedge\bigvee_{\ell\leqslant n}(\ell\leqslant i\wedge\Gamma^\ell(i)=\rho).$$

This gives the union over all $\ell\leqslant n$. Clearly $\Upsilon\in\mathscr{C}$ and $\Upsilon(i)\subseteq\overline{F}(i)$ for all $i\in\mathbb{N}$.

It remains to show that $\overline{F}(i)\subseteq\Upsilon(i)$. To do this, it will suffice to find $\ell\leqslant n$, $z_0\in[n]$, and $1\leqslant d_1<d_2<\cdots<d_\ell$ such that

$$F_4^\ell(i,z_0,d_1,\ldots,d_\ell)=\overline{F}(i).$$

Note that in this part of the proof, the properties of $\mathscr{C}$ are no longer relevant: we make external observations about functions already shown to be in $\mathscr{C}$.

We begin with two claims, whose proofs are trivial.

Claim 1. *If* $\rho,\sigma\in\mathscr{B}_n$ *then*

$$\langle y,z\rangle\in\sigma \text{ implies } \rho^{-1}(y)\subseteq(\rho\circledast\sigma)^{-1}(z).$$

Claim 2. *If* $\rho,\sigma\in\mathscr{B}_n$ *and*

$$(\rho\circledast\sigma)^{-1}(z)\subseteq\rho^{-1}(y)$$

then

$$\rho\circledast\tau(\sigma,y,z)=\rho\circledast\sigma.$$

Starting from $F:\mathbb{N}\to\mathscr{T}_n^\sharp$, we choose a $z_0\in[n]$ and fix a sequence $d_1,d_2\ldots\leqslant i$ as follows. z_0 is any element of $[n]$. We define $F_1:\mathbb{N}\to\mathscr{T}_n$ and $P_2:\mathbb{N}\to[n]$ exactly as above.

Let d_1 be the least $k>0$ such that $k\leqslant i$ and

$$\overline{F}(k-1)^{-1}(P_2(k))\supsetneq\{z_0\}.$$

If no such k exists, stop.

Thereafter, let d_m be the least $k > d_m$ such that $k \leqslant i$ and

$$\overline{F}(k-1)^{-1}(P_2(k)) \supsetneq \overline{F}((d_m)-1)^{-1}(P_2(d_m)).$$

If no such k exists, stop.

Since

$$\{z_0\} \subsetneq F((d_1)-1)^{-1}(d_1) \subsetneq F((d_2)-1)^{-1}(d_2) \subsetneq \cdots \subseteq [n],$$

the sequence must be finite, indeed of length $\leqslant n$.

For a given $i \in \mathbb{N}$, let $\ell \leqslant n$ be such that $d_\ell \leqslant i$ and either the sequence stops at d_ℓ or $d_{\ell+1} > i$.

We have the following property:

Claim 3. *For* $0 < j < d_1$,

$$\overline{F}(j-1)^{-1}(P_2(j)) = \{z_0\}$$

and for all $d_m < j < d_{m+1}$ $(0 < m < \ell)$ *or* $j > d_m$ $(m = \ell)$,

$$\overline{F}(j-1)^{-1}(P_2(j)) = \overline{F}((d_m)-1)^{-1}(P_2(d_m)).$$

Proof. For the case $d_m < j < d_{m+1}$ $(0 < m < \ell)$ or $j > d_m$ $(m = \ell)$, we observe that $\langle P_2(k-1), P_2(k)\rangle \in F_1(k-1) \subseteq F(k-1)$. Then a simple induction based on Claim 1 shows that

$$\begin{aligned}\overline{F}(j-1)^{-1}(P_2(j)) &\supseteq (\overline{F}((d_m)-1) \circledast (F_1(d_m) \circledast \cdots \circledast F_1(j-1)))^{-1}(z_0)\\ &\supseteq (\overline{F}((d_m)-1))^{-1}(P_2(d_m)).\end{aligned}$$

On the other hand

$$\overline{F}(j-1)^{-1}(P_2(j)) \supsetneq \overline{F}((d_m)-1)^{-1}(P_2(d_m))$$

would violate the definition of d_{m+1}. The case $0 < j < d_1$ is very similar. This ends the proof of Claim 3. □

Let $F_3 \to \mathscr{B}_n$ be exactly as above.

Claim 4. *For* $0 < j < j+1 < d_1$, $d_m < j < j+1 < (d_{m+1}) - 1$ $(0 < m < \ell)$, *and* $j < d_\ell$,

$$\overline{F}(j) = \overline{F}(j-1) \circledast F_3(j+1).$$

Furthermore $d_1 \geqslant 2$ *implies that*

$$F(0) = F_3(1).$$

Proof. By Claim 3,

$$\overline{F}(j)^{-1}(P_2(j+1)) = \{z_0\} = \overline{F}(j-1)^{-1}(P_2(j))$$

or

$$\overline{F}(j)^{-1}(P_2(j+1)) = \overline{F}((d_m)-1)^{-1}(P_2(d_m)) = \overline{F}(j-1)^{-1}(P_2(j)).$$

Therefore, applying Claim 2, with

$$\rho = \overline{F}(j-1),$$
$$\sigma = F(j),$$
$$y = P_2(j),$$
$$z = P_2(j+1),$$

we see that

$$\overline{F}(j-1) \circledast \tau(F(j), P_2(j), P_2(j+1)) = \overline{F}(j-1) \circledast F(j).$$

That is

$$\overline{F}(j-1) \circledast F_3(j+1) = \overline{F}(j).$$

In the case $d_1 \geqslant 2$,

$$F(0)^{-1}(P(1)) = \{z_0\}$$

and therefore

$$\tau(F(0), z_0, P(1)) = F(0).$$

This completes the proof of Claim 4. □

Easy inductions then show

Claim 5. *For* $0 < j < d_1$,

$$\overline{F}(j-1) = F_3(1) \circledast \cdots \circledast F_3(j)$$

and, thereafter, for $d_m \leqslant j < d_{m+1}$ $(0 < m < \ell)$ *or* $j > d_m$ $(m = \ell)$,

$$\overline{F}(j-1) = \overline{F}((d_m)-1) \circledast F_3((d_m)+1) \circledast \cdots \circledast F_3(j).$$

From Claim 5, it follows immediately that

$$F_4^{\ell}(i) = \overline{F}(i).$$

Therefore, since $F_4^{\ell}(i, z_0, d_1, \ldots, d_{\ell}) \subseteq \Upsilon(i)$,

$$\Upsilon(i) = \overline{F}(i).$$

Lemma 9 is thus proved. □

All the steps necessary for the proof of the main theorem (Theorem 1) have now been presented.

7. Bel'tyukov's stack register machines

In this section we introduce the deterministic stack register machine (SRM) and explain how it computes a function or recognises a relation. This done, we introduce the nondeterministic SRM but then apply Theorem 1 to show that the nondeterminism can be eliminated.

Deterministic SRMs

To make the following definitions and results as general as possible we use $\mathscr{F}$ to sand for some set of functions over the natural numbers $\mathbb{N}$. These act as oracles. Our principal example is $\mathscr{F} = \{+, \cdot\}$, in which case we write SRM[+, ·], etc.

Definition 13. An SRM $[\mathscr{F}]$ calculating an a-ary function has a input registers with contents $x_1, \ldots, x_a$; b stack registers with contents $t_b, \ldots, t_0$; and a working register containing r. The instructions of the machine can be the following:

(i) $r := z$, where z has the form x_i, t_i, r or 0;
(ii) $t_i := t_i + 1$;
(iii) *if* $f(z_1, \ldots, z_n) = z_{n+1}$ *then go to* L_i *else go to* L_j;
(iv) *halt*,

where: z and the z_i have the form x_m, t_m, r or 0; L_i, L_j label instructions; $f \in \mathscr{F}$. A program is a sequence of consecutively numbered instructions. An important restriction is that, for $0 \leqslant i \leqslant b$, there is at most one instruction $t_i := t_i + 1$ affecting t_i.

Before execution the inputs are placed in the input registers. Non-input registers are initialised to zero. The instruction $t_i := t_i + 1$ has the side-effect of setting

$$t_j := 0$$

for all $j < i$. The operator of the program is otherwise as one would expect. The value computed is t_b at *halt*.

Remarks. The proof of Lemma 11 below, gives some examples of programs. During execution, the input registers are like a block of high-index stack registers, for none of which is there an instruction of type (ii).

If the value we want ends up in the bottom register t_0 (or any other register), we can add a new top stack register t_{b+1}, and to the instructions add an outer loop which keeps incrementing t_{b+1}, each time running through the original program in its entirety, until we have the same value in t_{b+1} as t_0. (This assumes that we can test for equality between registers, but this is nearly always possible.) Therefore requiring output from the top stack-register is not a significant restriction (even when we come to consider space-bounded complexity classes).

We often find it easier to use machines of what we will call type $\mathrm{SRM}_{\mathscr{G}}$; these are described in [3] under the name SRM_f (in a slightly restricted setting). Subject to very weak conditions, they have the same computational power.

Definition 14. For $\mathscr{G}$ a class of functions, an $\mathrm{SRM}_{\mathscr{G}}$ calculating an a-ary function has input registers $x_1,\ldots,x_a$; stack registers $t_b,\ldots,t_0$; and a working register r. The machine operates in a loop of the form

$$L:\left\{\begin{array}{lll}
\underline{\textit{if}} & \Psi_0(\vec{x},\vec{t},r)=0 \ \underline{\textit{then}} & (r:=\chi_0(\vec{x},\vec{t},r);\ t_0:=t_0+1;\ \underline{\textit{go to}}\ L)\\
\underline{\textit{else if}} & \Psi_1(\vec{x},\vec{t},r)=0 \ \underline{\textit{then}} & (r:=\chi_1(\vec{x},\vec{t},r);\ t_1:=t_1+1;\ \underline{\textit{go to}}\ L)\\
 & \vdots & \\
\underline{\textit{else if}} & \Psi_b(\vec{x},\vec{t},r)=0 \ \underline{\textit{then}} & (r:=\chi_b(\vec{x},\vec{t},r);\ t_b:=t_b+1;\ \underline{\textit{go to}}\ L)\\
\underline{\textit{else}} & & (r:=\chi_{b+1}(\vec{x},\vec{t},r);\ \underline{\textit{halt}})
\end{array}\right\}$$

where $\Psi_0,\ldots,\Psi_b,\chi_0,\ldots,\chi_{b+1}\in\mathscr{G}$. Initialisation is as before and the instruction $t_i:=t_i+1$ still has the side-effect of setting $t_j:=0$, for all $j<i$.

As in [8], we have:

Definition 15. For $\phi,\varrho:\mathbb{N}\to\mathbb{N}$,

$$\mathrm{Space}[\mathscr{F}](\phi,\varrho)$$

is the collection of functions f over $\mathbb{N}$ such that for some SRM, for every input $x_1,\ldots,x_a$ (a being the arity of f):

1. the SRM eventually halts;
2. upon halting $t_b=f(x_1,\ldots,x_a)$;
3. at all times during the computation,
   (a) $t_i\leqslant\phi(\max\{x_1,\ldots,x_a\})$ for $i\leqslant k$, and
   (b) $r\leqslant\varrho(\max\{x_1,\ldots,x_a\})$.

We may replace ϕ and/or ϱ by whole classes of nondecreasing functions, such as $n^{O(1)}$, the class of polynomial bounds.

We define $\mathrm{Space}_{\mathscr{G}}(\phi,\varrho)$ etc. similarly to $\mathrm{Space}[\mathscr{F}](\phi,\varrho)$ in Definition 15.

Definition 16. A relation $R\subseteq\mathbb{N}^a$ is in $\mathrm{SRM}[\mathscr{F}](\phi,\varrho)$ just if there is an $f\in\mathrm{Space}[\mathscr{F}](\phi,\varrho)$ such that

$$f(x_1,\ldots,x_a)=0 \quad \text{iff } \langle x_1,\ldots,x_a\rangle\in R.$$

Remark. SRMs are especially apt for computing primitive recursive functions. In particular, the levels

$$\mathscr{E}^0,\mathscr{E}^1,\mathscr{E}^2,\ldots$$

of the Grzegorczyk hierarchy (see [18]) can be expressed as complexity classes of SRMs. There are corresponding classes of relations, and it is a problem of long standing whether one can separate

$$\mathscr{E}^0_*, \mathscr{E}^1_*, \mathscr{E}^2_*,$$

the three smallest of these. It is known that

$$\Delta^{\mathbf{N}}_0 \subseteq \mathscr{E}^0_* \quad \text{and} \quad \mathscr{E}^2_* = \text{DLINTIME}.$$

The latter is due to Ritchie [17]. The former inclusion is not known to be strict and our chief interest lies in this possible gap between $\Delta^{\mathbf{N}}_0$ and $\mathscr{E}^0_*$. One will readily observe that $\mathscr{E}^0_*$ is closed under the summations treated here, so a negative result $G\text{-}\Delta^{\mathbf{N}}_0 \neq \Delta^{\mathbf{N}}_0$ would show $\Delta^{\mathbf{N}}_0 \subsetneq \mathscr{E}^0_*$ and hence LTH $\subsetneq$ LINSPACE.

Facts (Bel'tyukov [3]).

$$\mathscr{E}^0 = \text{Space}[\underline{id}](n + \text{O}(1), n + \text{O}(1)),$$

$$\mathscr{E}^1 = \text{Space}[\underline{id}](n \cdot \text{O}(1), n \cdot \text{O}(1)),$$

$$\mathscr{E}^2 = \text{Space}[\underline{id}](n^{\text{O}(1)}, n^{\text{O}(1)})$$

and hence

$$\mathscr{E}^0_* = \text{SRM}[\underline{id}](n + \text{O}(1), n + \text{O}(1)),$$

$$\mathscr{E}^1_* = \text{SRM}[\underline{id}](n \cdot \text{O}(1), n \cdot \text{O}(1)),$$

$$\mathscr{E}^2_* = \text{SRM}[\underline{id}](n^{\text{O}(1)}, n^{\text{O}(1)}).$$

Furthermore

$$\Delta^{\mathbf{N}}_0 = \text{SRM}[+, \cdot](n^{\text{O}(1)}, 0).$$

Fact 1 (Paris and Wilkie [16]).

$$S_{k+1}\text{-}\Delta^{\mathbf{N}}_0 = \text{SRM}[+, \cdot](n^{\text{O}(1)}, k).$$

Fact 2 (Clote [8]). *For all* $k \geqslant 4$

$$\text{SRM}[+, \cdot](n^{\text{O}(1)}, k) = \text{ALINTIME}.$$

Lemma 10. *Suppose that* Φ, Γ *are classes of nondecreasing functions on* $\mathbb{N}$ *such that*:

1. $(\forall k \in \mathbb{N})(\exists \phi \in \Phi)(\forall x \in \mathbb{N})(\phi(x) \geqslant x + k)$;
2. $(\forall \varrho_1, \varrho_2 \in \Gamma)(\exists \varrho_3 \in \Gamma)(\forall x \in \mathbb{N})(\varrho_1(x) \leqslant \varrho_3(x) \wedge \varrho_2(x) \leqslant \varrho_3(x))$;

3. $(\forall \varrho_1 \in \Gamma, \phi \in \Phi)(\exists \varrho_2 \in \Gamma)(\forall x \in \mathbb{N})(\varrho_1(\phi(x)) \leqslant \varrho_2(x))$;
4. $(\forall \varrho \in \Gamma)(\exists \phi \in \Phi)(\forall x \in \mathbb{N})(\varrho(x) \leqslant \phi(x))$.

And suppose that there is an eq $\in \mathscr{F}$ *such that for some* $\xi_1, \ldots, \xi_n, \xi_{n+1} \in \{z_1, z_2\} \cup \{0, 1\}$

$$\mathrm{eq}(\xi_1, \ldots, \xi_n) = \xi_{n+1} \quad \textit{iff } z_1 = z_2.^6 \tag{6}$$

It follows that

$$Space_{Space[\mathscr{F}](\Phi,\Gamma)}(\Phi, \Gamma) = Space[\mathscr{F}](\Phi, \Gamma).$$

Proof. (This is a special case of [9, Lemma II.3].) We give only a suggestion of the proof; it is essentially as for the first step in the proof of [3, Theorem 1] (or see [8] on "normalisation"), but having cast our definition slightly more generally, we have been correspondingly more explicit about the conditions that make the proof work.

Calculations of the χ_i can easily be implemented on new stack registers of low index introduced specially for that purpose. It is therefore easy to see that SRM$_{\mathscr{G}}$s can be simulated by standard SRMs.

The converse is less trivial and relies on the fact that there is at most one instruction of type (ii) for each stack register t_i. Because of the side-effect

$$t_j := 0 \quad (j < i)$$

it is always known which was the last stack register to be changed. Between executions of instructions of type (ii) there are only branchings and changes to the work register r. If r changes it can only assume one of the unchanged values $\{x_1, \ldots, x_a, t_b, \ldots, t_0\}$. If execution is to terminate, there must be no loops. Therefore, a type (iii) instruction can be applied at most $a + b + 2$ times between executions of instructions of type (ii) (or before the final *halt*). It is relatively easy to find functions $\chi_0, \ldots, \chi_{b+1}$, which capture the possibilities. □

Lemma 11. *Under the same conditions as Lemma* 10,

$$SRM[\mathscr{F}](\Phi, \Gamma) \textit{ is } \Delta_0\textit{-closed.}$$

Proof. The program

1. $t_0 := t_0 + 1$
2. *halt*.

computes the constant function 1. Therefore the program

1. $t_0 := t_0 + 1$;
2. *if* $\mathrm{eq}(\xi'_1, \ldots, \xi'_n) = \xi'_{n+1}$ *then go to* 4;
3. $t_1 := t_1 + 1$;
4. *halt*.

[6] For example, if $+ \in \mathscr{F}$, then $z_1 + 0$ is such an eq, with $z_{n+1} = z_2$.

recognises $x_1 = x_2$, where ξ_i' is x_1 if ξ_i is z_1, x_2 if ξ_i is z_2, 0 if ξ_i is 0, and t_0 if ξ_i is 1. This allows us to simulate branching instructions of the form *if* $z_1 = z_2$ *then* ..."[7]

The binary relation $x_1 \leqslant x_2$ is recognised, in space bounds $(id, 0)$, by

1. *if* $t_0 = x_1$ *then goto* 6;
2. *if* $t_0 = x_2$ *then goto* 5;
3. $t_0 := t_0 + 1$;
4. *if* $t_0 = t_0$ *then goto* 1;
5. $t_1 := t_1 + 1$;
6. *halt*.

For Δ_0-definability:

(a) *Boolean operations*: Are very easy.

(b) *Bounded quantification*: For bounded existential quantification, we have:

$$L: \left\{ \begin{array}{ll} \underline{if} & t_1 = 0 \wedge t_2 = 0 \wedge t_0 < x_i \wedge \neg R(t_0, \vec{x}) \\ & \underline{then}\ (r := r;\ t_0 := t_0 + 1;\ \underline{go\ to}\ L) \\ \underline{else\ if} & t_1 = 0 \wedge t_2 = 0 \wedge t_0 < x_i \wedge R(t_0, \vec{x}) \\ & \underline{then}\ (r := r;\ t_1 := t_1 + 1;\ \underline{go\ to}\ L) \\ \underline{else\ if} & t_1 = 0 \wedge t_2 = 0 \wedge t_0 = x_i \wedge \neg R(t_0, \vec{x}) \\ & \underline{then}\ (r := r;\ t_2 := t_2 + 1;\ \underline{go\ to}\ L) \\ \underline{else} & (r := r; \quad \underline{halt}) \end{array} \right\}$$

Bounded universal quantification is equally easy (or follows from closure under complementation).

(c) *Explicit transformation*: This is also straightforward. To obtain natural number constants other than 0 and 1, we use the appropriate number of applications of the successor function starting from 0. The successor function has program

1. *if* $t_0 = t_1$ *then go to* 5;
2. *if* $t_0 = x$ *then go to* 7;
3. $t_0 := t_0 + 1$;
4. *goto* 1;
5. $t_1 := t_1 + 1$;
6. *goto* 1;
7. *halt*.

None of these programs use the work register. The stack registers are always bounded by $\lambda x \cdot x + k$, for some k. □

[7] There is a slight subtlety here. To know if $z_1 = z_2$ one must perform a subcomputation. However, because one must subsequently resume the original flow, this is not a straightforward case of composition (which is easy). The problem is that both the interrupting and interrupted computations affect the work register. Fortunately one can store the value of the work register without difficulty in a new low-index stack register and zero the work register; one then performs the interrupting computation on stack registers of even lower index, restoring the work register to its stored value before continuing.

Theorem 2. *Under the same conditions as Lemma* 10, *for all* $n \in \mathbb{N}$, SRM$[\mathscr{F}](\Phi,\Gamma)$ *is closed under summation mod* S_n *if and only if* SRM$[\mathscr{F}](\Phi,\Gamma)$ *is closed under summation mod* $\mathscr{B}_n$.

Proof. Theorem 1 + Lemma 11. □

Nondeterministic SRMs

In [8], Clote gives the definition of the nondeterministic SRM (NSRM):

Definition 17. A *nondeterministic stack register machine* over $\mathscr{F}$, denoted NSRM$[\mathscr{F}]$, is defined by extending the branching instruction (iii) of Definition 13 to

(iii)′ *if* $f(z_1,\ldots,z_n) = z_{n+1}$ *then go to* $L_{a_1},\ldots,L_{a_r}$ *else go to* $L_{b_1},\ldots,L_{b_s}$.

If $f(z_1,\ldots,z_n) = z_{n+1}$, the machine may jump to any of the instructions labelled $L_{a_1},\ldots,L_{a_r}$. If $f(z_1,\ldots,z_n) \neq z_{n+1}$, it may jump to any of the instructions $L_{b_1},\ldots,L_{b_s}$.

For a given input $x_1,\ldots,x_a$, there will now be a *computation tree* instead of a single computation. This tree will have branch points corresponding to the instructions of type (iii)′. The nodes in the computation tree are the configurations $(L_j,x_1,\ldots,x_1,t_b,\ldots,t_0,r)$ which can be reached from the initial configuration $(L_0,x_1,\ldots,x_1,0,\ldots,0,0)$, which later forms the root node: L_j represents the next instruction to be executed with L_0 the initial instruction.

Definition 18. An input $x_1,\ldots,x_a$ is *accepted* by an NSRM if:

(a) The computation tree contains only finitely many different configurations.

(b)There is a computation, i.e. single branch in the computation tree which has a leaf node $(L_h,x_1,\ldots,x_1,t_b,\ldots,t_0,r)$, where L_h is a *halt* instruction, and where $t_b = 0$.

Complexity classes for NSRMs are defined in much the same way as those for SRMs (Definition 15):

Definition 19. An a-ary relation R is in NSRM$[\mathscr{F}](\Phi,\Gamma)$ if there is an NSRM such that:

(a) For all inputs $x_1,\ldots,x_a$, all the configurations in the computation tree with root $(L_0,x_1,\ldots,x_1,0,\ldots,0,0)$ satisfy the bounds (Φ,Γ).

(b) The machine accepts input $x_1,\ldots,x_a$ if and only if $\langle x_1,\ldots,x_a\rangle \in R$.

For the rest of the section, we confine ourselves to the complexity classes NSRM $[\mathscr{F}]$ (Φ,k). This means that the work register r is only permitted to take values in $[k+1] = \{0,\ldots,k\}$. The following lemma will be useful.

Lemma 12. *Under the same conditions as Lemma* 10, SRM $[\mathscr{F}]$ (Φ,k) *is closed under summation mod* $\mathscr{B}_{k+1}$.

Proof. If $F:\mathbb{N}^a \to S_{k+1}$ and $F \in \mathscr{C}$ then (Definition 9), for all pairs

$$\langle u,v\rangle \in [k+1]\times[k+1],$$

the a-ary relation

$$F^{-1}(\langle u,v\rangle)$$

is in $\mathscr{C}$. Let

$$\Psi(\vec{x},y,z) \quad \text{iff} \bigvee_{u,v\in[k+1]} (y=u\wedge z=v\wedge \vec{x}\in F^{-1}(\langle u,v\rangle)).$$

For $y \in [k+1]$, the program

$$L:\left\{\begin{array}{ll}\underline{if} & y\leqslant k\wedge\neg\Psi_0(\vec{x},y,t_0)\ \underline{then}\ (t_0:=t_0+1;\ \underline{go\ to}\ L)\\ \underline{else} & (\underline{halt})\end{array}\right\}$$

calculates $(F(\vec{x}))(y)$. The program

$$L:\left\{\begin{array}{lll}\underline{if} & (t_3=0\wedge t_2=0\wedge t_1>0\wedge t_0\leqslant x_1) & \\ & \underline{then}\ (r:=(F(t_0,x_2,\ldots,x_a))(r); & t_0:=t_0+1;\ \underline{go\ to}\ L)\\ \underline{else\ if} & (t_3=0\wedge t_2=0\wedge t_1=0\wedge r\neq u) & \\ & \underline{then}\ (r:=u; & t_1:=t_1+1;\ \underline{go\ to}\ L)\\ \underline{else\ if} & (t_3=0\wedge t_2=0\wedge t_1>0\wedge t_0>x_1\wedge r=v) & \\ & \underline{then} & (t_2:=t_2+1;\ \underline{go\ to}\ L)\\ \underline{else\ if} & (t_3=0\wedge t_2=0\wedge t_1>0\wedge t_0>x_1\wedge r\neq v) & \\ & \underline{then} & (t_3:=t_3+1;\ \underline{go\ to}\ L)\\ \underline{else} & & (\underline{halt})\end{array}\right\}$$

recognises $\overline{F}^{-1}(\langle u,v\rangle)$. The summation is with respect to x_1 with $x_2,\ldots,x_a$ as parameters. At $\underline{halt}$, $t_3=0$ indicates acceptance. □

Corollary 2. *Under the same conditions as Lemma* 10,

$$\mathscr{B}_{k+1}\text{-}\Delta_0[\mathscr{F}]\subseteq \mathrm{SRM}[\mathscr{F}](\Phi,k).$$

Proof. Lemmas 11 and 12. □

In this case, a restricted application of Clote's "normalisation" lemma shows that the NSRM's program can be replaced by one of the form

$$L:\left\{\begin{array}{l}
\underline{\textit{if}}\ \Psi_{0,0}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=0;\ t_0:=t_0+1;\ \underline{\textit{go to}}\ L)\\
\qquad\qquad\qquad\qquad\qquad\qquad\vdots\\
\underline{\textit{if}}\ \Psi_{0,k}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=k;\ t_0:=t_0+1;\ \underline{\textit{go to}}\ L)\\
\qquad\qquad\qquad\qquad\qquad\qquad\vdots\\
\underline{\textit{if}}\ \Psi_{i,v}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=v;\ t_i:=t_i+1;\ \underline{\textit{go to}}\ L)\\
\qquad\qquad\qquad\qquad\qquad\qquad\vdots\\
\underline{\textit{if}}\ \Psi_{b,k}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=k;\ t_b:=t_b+1;\ \underline{\textit{go to}}\ L)\\
\underline{\textit{if}}\ \Psi_{h,0}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=0;\ \underline{\textit{halt}})\\
\qquad\qquad\qquad\qquad\qquad\qquad\vdots\\
\underline{\textit{if}}\ \Psi_{h,k}(\vec{x},\vec{t},r)\ \underline{\textit{then}}\ (r:=k;\ \underline{\textit{halt}})
\end{array}\right\}$$

where $\Psi_{0,0}, \Psi_{0,1},\ldots,\Psi_{0,k},\Psi_{1,0},\ldots,\Psi_{b,k},\Psi_{h,0},\Psi_{h,k}\in \mathrm{SRM}[\mathscr{F}](\Phi,k)$. Because these relations need not be disjoint, the program is still nondeterministic.

The full proof of this is considerably more complicated than the deterministic equivalent, and we do not attempt to reproduce Clote's argument.

Theorem 3. *Under the same conditions as Lemma* 10, *and provided that the functions in* Φ *are themselves conputable by* SRM[$\mathscr{F}$]*'s running in bounds* (Φ,k),

$$\mathrm{NSRM}[\mathscr{F}](\Phi,k)=\mathrm{SRM}[\mathscr{F}](\Phi,k).$$

Proof. Here we sketch an alternative version of Clote's "looping" lemma for NSRMs, which he gives in full for the case $k=1$.

Let $\phi\in\mathrm{SRM}[\mathscr{F}](\Phi,k)$ be the bound on the stack registers.

Taking the normalised program above, first define $F_0\in\mathrm{NSRM}\ [\mathscr{F}](\Phi,k)$ by

$$t_0\in F_0^{-1}(\langle u,v\rangle)\quad\text{iff}\quad \Phi_{0,v}(\vec{x},\vec{t},u).$$

$F_0:\mathbb{N}\to\mathscr{B}_{k+1}$ and so, by Lemma 12 and Theorem 2, $\overline{F}_0\in\mathrm{SRM}[\mathscr{F}](\Phi,k)$. Next, for $1\leqslant i\leqslant b$ and $i=h$, define $\Psi'_{i,v}(\vec{x},t_b,\ldots,t_1,r)$ as

$$\Psi_{i,v}(\vec{x},t_b,\ldots,t_1,0,r)$$

$$\vee$$

$$(\exists t_0\leqslant\phi(\max(\vec{x})))\left(\bigvee_{w\in[k+1]}\left(\begin{array}{c}\langle r,w\rangle\in((\overline{F}_0)^{-1}(t_0))\\ \wedge\\ \Psi_{i,v}(\vec{x},t_b,\ldots,t_1,t_0,w)\end{array}\right)\right).$$

Using the various known closure properties, it is easy to see that

$$\Psi'_{i,v}\in\mathrm{SRM}[\mathscr{F}](\Phi,k).$$

The use of $\phi(\max(\vec{x}))$ can be thought of as composition with ϕ.

$\Psi'_{i,v}$ covers all the possibilities for the next move affecting $t_b,\ldots,t_1$ to be

$$t_i:=t_i+1.$$

Because changes to higher stack registers have the side effect $t_0 := 0$, any intermediate changes to t_0 must start from $t_0 = 0$. At the point where we jump out of such a succession of increments to t_0, $\overline{F_0}(t_0)$ allows us to recover the possible values in the work register, but nothing more. Furthermore, the machine must jump out before hitting the bound $\phi(\max(\vec{x}))$. For these reasons, the program

$$L: \left\{ \begin{array}{l} \underline{\textit{if}}\ \Psi'_{1,0}(\vec{x}, t_b, \ldots, t_1, r)\ \underline{\textit{then}}\ (r := 0;\ t_1 := t_1 + 1;\ \underline{\textit{go to}}\ L) \\ \qquad\qquad \vdots \\ \underline{\textit{if}}\ \Psi'_{i,v}(\vec{x}, t_b, \ldots, t_1, r)\ \underline{\textit{then}}\ (r := v;\ t_i := t_i + 1;\ \underline{\textit{go to}}\ L) \\ \qquad\qquad \vdots \\ \underline{\textit{if}}\ \Psi'_{b,k}(\vec{x}, t_b, \ldots, t_1, r)\ \underline{\textit{then}}\ (r := k;\ t_b := t_b + 1;\ \underline{\textit{go to}}\ L) \\ \underline{\textit{if}}\ \Psi'_{h,0}(\vec{x}, t_b, \ldots, t_1, r)\ \underline{\textit{then}}\ (r := 0;\ \underline{\textit{halt}}) \\ \qquad\qquad \vdots \\ \underline{\textit{if}}\ \Psi'_{h,k}(\vec{x}, t_b, \ldots, t_1, r)\ \underline{\textit{then}}\ (r := k;\ \underline{\textit{halt}}) \end{array} \right\}$$

recognises the same relation as before and runs within the required bounds. The register t_0 is no longer used.

By repeating this step, we eventually arrive at

$$L: \left\{ \begin{array}{l} \underline{\textit{if}}\ \Psi''_{b,0}(\vec{x}, t_b, r)\ \underline{\textit{then}}\ (r := 0;\ t_b := t_b + 1;\ \underline{\textit{go to}}\ L) \\ \qquad\qquad \vdots \\ \underline{\textit{if}}\ \Psi''_{b,k}(\vec{x}, t_b, r)\ \underline{\textit{then}}\ (r := k;\ t_b := t_b + 1;\ \underline{\textit{go to}}\ L) \\ \underline{\textit{if}}\ \Psi''_{h,0}(\vec{x}, t_b, r)\ \underline{\textit{then}}\ (r := 0;\ \underline{\textit{halt}}) \\ \qquad\qquad \vdots \\ \underline{\textit{if}}\ \Psi''_{h,k}(\vec{x}, t_b, r)\ \underline{\textit{then}}\ (r := k;\ \underline{\textit{halt}}) \end{array} \right\}$$

with the $\Psi''_{b,0}, \ldots, \Psi''_{h,k} \in \mathrm{SRM}[\mathscr{F}](\Phi, k)$. This machine accepts just if

$$\bigvee_{v \in [k+1]} (\Psi''_{h,v}(\vec{x}, t_b, 0))$$

which expresses the possibility of a *halt* before any increment of t_b. □

Corollary 3.

$$\mathrm{NSRM}[+,\cdot](n^{\mathrm{O}(1)}, k) = \mathrm{SRM}[+,\cdot](n^{\mathrm{O}(1)}, k) = S_{k+1}\text{-}\Delta_0^{\mathrm{N}}.$$

Proof. It is easy to verify that the conditions for Theorem 3 are met and that the observation

$$\Psi_{0,0}, \Psi_{0,1}, \ldots, \Psi_{0,k}, \Psi_{1,0}, \ldots, \Psi_{b,k}, \Psi_{h,0}, \Psi_{h,k} \in \mathrm{SRM}[\mathscr{F}](\Phi, k)$$

in its proof can in this case be replaced by

$$\Psi_{0,0}, \Psi_{0,1}, \ldots, \Psi_{0,k}, \Psi_{1,0}, \ldots, \Psi_{b,k}, \Psi_{h,0}, \Psi_{h,k} \in \mathscr{B}_{k+1}\text{-}\Delta_0[\mathscr{F}].$$

We can then rely on Corollary 2. □

Indeed, for exactly the same reasons:

Corollary 4. *If there is an* eq $\in \mathscr{F}$ *satisfying* (6) *above then*

$$\mathrm{NSRM}[\mathscr{F}](n^{O(1)},k) = \mathrm{SRM}[\mathscr{F}](n^{O(1)},k) = S_{k+1}\text{-}\Delta_0[\mathscr{F}].$$

8. Concluding remarks

What emerges most clearly from this paper is the power of bounded quantification. In part of [8], Clote applies the techniques of [2], where Barrington showed the power of width-5 branching programs, to stack register machines whose work register can take at least 5 values. In other parts the work register is only permitted $1, 2, 3$ or 4 values. The results presented here also apply at these low levels; we might hope, therefore, to apply the new work to branching programs of width below 5. Unfortunately, this seems to be hard. Bounded quantification is not easily available with branching programs. To be sure, what Barrington shows is precisely that logical connectives (and hence bounded quantification) can be simulated, but the simulation requires a width of at least 5. What causes difficulty below this level? Just as with boolean circuits, defining an infinite set using branching programs requires a family of programs, one for each size of input. These families are basically nonuniform (each program may be completely unrelated to the others in the family) and imposing uniformity, say to eliminate nonrecursive sets, means imposing restrictions at a higher level than that of the circuits/programs themselves. On the other hand the individual programs are very inflexible; their "control flow" is the same for every input. By contrast, the sets in $\Delta_0^{\mathbf{N}}$, expressed by single formulae, are inherently uniform. Compensation comes in the form of the bounded quantifiers, which provide for wide-ranging searches, in effect allowing two inputs of the same size to be handled in highly divergent ways. Thus branching programs are powerful in being nonuniform but weak in having a fixed control flow for all inputs of the same size, whereas $\Delta_0^{\mathbf{N}}$ is weak in being uniform but powerful in having flexible control flow. At low levels the opposition nonuniform/inflexible vs. uniform/flexible seems to be significant.

On a different point, we note that the proofs abvoe do not depend on the presence of addition or multiplication, despite the fact that the intended application is to arithmetic. (The reader may recall similar independencies in the work of Ajtai [1].) By contrast, Clote, in proving that $S_5\text{-}\Delta_0^{\mathbf{N}} = \mathrm{ALINTIME}$, does use addition and multiplication (they make it possible to code strings). Although it is generally felt that $\mathrm{LTH} \subsetneq \mathrm{ALINTIME}$, we may therefore reasonably say that $+$ and $\cdot$ have yet to be fully understood.

Lastly, in [8] Clote introduces a notion of nondeterministic summation, intended to correspond to the nondeterminism of SRMs and indicated by the prefix N. Without giving definition, we remark that since the first version of this paper, progress has been

made. Combining the results here with those in [8, 10–13], we now know that

$$
\begin{array}{lcl}
N\mathscr{T}_3^{\sharp}\text{-}\Delta_0^{\mathbf{N}} & = & S_5\text{-}\Delta_0^{\mathbf{N}} = \text{ALTIME} \\
 & & \bigcup \\
N\mathscr{T}_4\text{-}\Delta_0^{\mathbf{N}} & = & S_3\text{-}\Delta_0^{\mathbf{N}} = \mathscr{B}_3\text{-}\Delta_0^{\mathbf{N}} \\
 & & \bigcup \\
N\mathscr{B}_2\text{-}\Delta_0^{\mathbf{N}} & = & S_2\text{-}\Delta_0^{\mathbf{N}} = \mathscr{B}_2\text{-}\Delta_0^{\mathbf{N}} \\
 & & \bigcup \\
 & & \Delta_0^{\mathbf{N}}
\end{array}
$$

Acknowledgements

I am indebted to the anonymous referee and also to Peter Clote; they pointed out many errors in the original version of this paper and suggested many improvements.

References

[1] M. Ajtai, Parity and the pigeonhole principle, in: S.R. Buss and P.J. Scott, eds., *Feasible Mathematics* (Birkhäuser, Basel, 1990) 1–24.

[2] D.A. Barrington, Bounded-width polynomial-size branching programs recognize exactly those languages in NC^1, *J. Comput. System Sci.* **38** (1989) 150–164.

[3] A.P. Bel'tyukov, A machine description and the hierarchy of initial Grzegorczyk classes, *J. Soviet Math.* **20** (1982) 2280–2289; translated from Zapiski Nauchnykh Seminarov Leningradskogo Otdeleniya Matematicheskogo Instituta im. V.A. Steklova AN SSR, Vol. 88 (1979) 30–46.

[4] J.H. Bennett, On spectra, Ph.D. Thesis, Department of Mathematics, Princeton University, 1962.

[5] S.R. Buss, Bounded arithmetic, *Studies in Proof Theory* (Bibliopolis, Napoli, 1986), revision of 1985 Princeton Ph.D. Thesis.

[6] A.K. Chandra, D.C. Kozen and L.J. Stockmeyer, Alternation, *J. ACM* **28** (1981) 114–133.

[7] A.H. Clifford and G.B. Preston, *The Algebraic Theory of Semigroups*, Mathematical Surveys, Vol. 7 (American Mathematical Society, Providence, RI, 2nd (1964) ed., 1961).

[8] P. Clote, Nondeterministic stack register machines, *Theoret. Comput. Sci.*, submitted.

[9] W.G. Handley, Some machine characterizations of classes close to $\Delta_0^{\mathbf{N}}$, Ph.D. Thesis, Department of Mathematics, University of Manchester, 1986.

[10] W.G. Handley, Deterministic summation modulo $\mathscr{B}_n$, the semigroup of binary relations on $\{0, 1, \ldots, n-1\}$, *Theoret. Compnt. Sci.*, accepted.

[11] W.G. Handley, Nondeterministic summation *mod* the semigroup of pregraphs over $\{0, 1, 2\}$ yields ALINTIME, December 1994.

[12] W.G. Handley, Nondeterministic summation modulo $\mathscr{T}_3$, the semigroup of functions on $\{0, 1, 2\}$, submitted.

[13] W.G. Handley, Nondeterministic summation modulo $\mathscr{T}_4$, the semigroup of functions on $\{0, 1, 2, 3\}$, September 1994.

[14] K. Harrow, The bounded arithmetic hierarchy, *Inform. and Control* **36** (1978) 102–117.

[15] J.B. Paris, W.G. Handley and A.J. Wilkie, Characterizing some low arithmetic classes, in: *Colloquia Mathematica Societatis Jànos Bolyai 44*, Hungary (1984) 353–364.

[16] J. Paris and A. Wilkie, Counting problems in bounded arithmetic, in: C.A. di Prisco, ed., *Proc. VIth Latin American, Logic Conf.* Caracus, Venezuela, Lecture Notes in Mathematics (Springer, Berlin, 1983) 317–340.

[17] R.W. Ritchie, Classes of predictably computable functions, *Trans. A.M.S.* **106** (1963).

[18] H.E. Rose. *Subrecursion: Functions and Hierarchies*, Oxford Logic Guides (Oxford University Press, Oxford, UK, 1984).
[19] R.M. Smullyan, *Theory of Formal Systems*, Annals of Mathematics Studies, Vol. 47 (Princeton University Press, Princeton, NJ, 1961).
[20] C. Wrathall, Complete sets and the polynomial-time hierarchy, *Theoret. Comput. Sci.* **3** (1977) 23–33.
[21] C. Wrathall, Rudimentary predicates and relative computation, *SIAM J. Comput.* **7** (1978) 194–209.